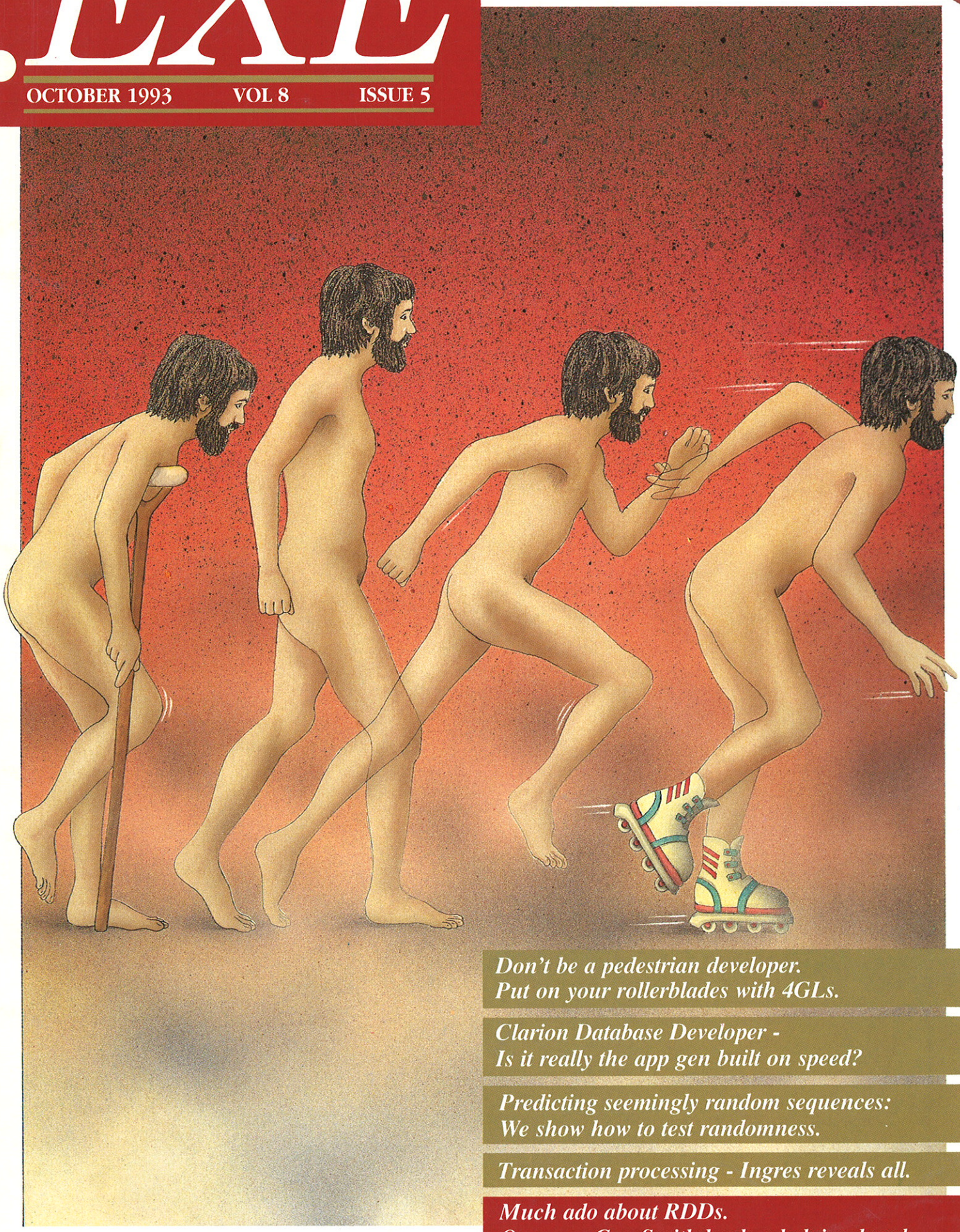
# EXE

£3.20

*The Software Developers' Magazine*

OCTOBER 1993 VOL 8 ISSUE 5

**SPECIAL REPORT**
**CA speaks on Visual Objects**



*Don't be a pedestrian developer.*
*Put on your rollerblades with 4GLs.*

*Clarion Database Developer -*
*Is it really the app gen built on speed?*

*Predicting seemingly random sequences:*
*We show how to test randomness.*

*Transaction processing - Ingres reveals all.*

*Much ado about RDDs.*
*Our man Guy Smith lends a helping hand.*

# Image Format Library

## Add TIFF, PCX, DCX, BMP, GIF, DIB, TARGA, WPG, WMF, EPS, PICT and JPEG support to your programs in less than 1 Hour!!

Image Format Library for C/C++/Pascal users provides an easy way to add powerful graphic imaging file format support to any application. Since both DOS & Windows versions are included, your application will work in both environments.


The libraries are designed to be easy to use, fast, and reliable. This simplicity comes from a design that hides all format complexity from the programmer. The libraries are fast because they employ modern optimization techniques- that include assembler code where necessary.

A single function call converts any of the file formats into a standardized format that can be used in any way by your application, even if a TIFF file has a PCX extension. It should take less than an hour to write import and export support for an application, and there will be no format-specific code, thus there is only one function to read and write, and everything defaults to the most commonly supported compression scheme. Compression type may also be specified.

A second-level interface is supported that provides a high degree of control over all functions as well as memory usage, plus the ability to read or write sub-regions, read and display in one step, and grab the screen directly to a compressed file. As the Windows Library is in DLL form, any Windows language can have instant image import/export, display, and printing features. Under DOS, all languages and compilers are supported. Image Format Library supports the following file formats: TIFF, PCX, DCX, BMP, GIF, DIB, TARGA, WPG, WMF, EPS, PICT, JPEG.
The VBX version is compatible with Visual Basic and Visual C++, and all features are implemented as properties of the control, rather than as function calls.

Available as a DLL or VBX

**£345**

+ Carriage + VAT

**ExpoTech**
CORPORATION

*The Leaders in Development Tool Distribution*

ExpoTech House, Western Road, Hove, Sussex. BN3 1AF.
Sales: +44-(0)273-749222 Fax: +44-(0)273-746446

Circle Info 060

Vol 8 Issue 5 .EXE October 1993

**Acting Editor:** Cliff Saran
**Production Manager:** Kate Adams
**Design & Layout:** Mark English
**Group Advertising Manager:** Sandra Inniss-Palmer
**Senior Advertising Executive:** Marc Warren
**Front Cover Illustration:** Alan Baker
**Printed by:** Chase Web Ltd, St Ives plc

## General

.EXE Magazine is independent and not affiliated to any vendor of hardware, software or services. It is published by Process Communications Ltd, 10 Barley Mow Passage, Chiswick, London W4 4PH.
Tel: (Advertising/Editorial/Production) 081 994 6477
(Subscriptions) 081 994 6477
Facsimile: 081 995 7775 Telex: 8811418 SPACES G
ISSN: 0268-6872

## Subscriptions

.EXE Magazine is a monthly journal for software developers. It is available only by subscription, at a cost of £35 per annum (12 issues) in the UK, £56 for 2 years. Overseas subscriptions cost £50 per annum, £70 for 2 years. 'A subscription implies that this journal will be sent to the subscriber until one of the three expires' - AG Macdonell. The magazine is published around the 26th of the month preceding the cover date. To subscribe or obtain details of overseas/academic rates, please call 081 994 6477, or write to The Subscriptions Manager, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We can invoice your company or take ACCESS and VISA cards. In case of a query about your subscription, please call 081 994 6477.
Back issues are available at a cost of £3.50 each - call our Chiswick office (081 994 6477) for a list of issues still in print.

## Editorial

Editorial enquiries should be addressed to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH or emailed to willw@dotexe.demon.co.uk. We welcome letters, opinions, suggestions and articles from our readers. Please write for a copy of our Contributors' Guide.
Information contained in .EXE is believed to be correct. If errors are found, we will endeavour to publish a clarification in the next available issue.
From time to time, we offer to copy the PC program code described in an article onto diskette for our readers. In this case, please send a blank, formatted disk with a self-addressed, prepaid mailer to the editorial address given above. We can copy both 5.25" and 3.5" disks.
The publishers can accept no liability for any consequences of using software distributed in this way.

## Advertising

If you are interested in advertising in this magazine, please write to the Display Advertising Manager/Recruitment Sales Manager (as appropriate), .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH, or call 081 994 6477 for details of our advertising rates.

## Pronunciation

.EXE Magazine rhymes with 'not sexy magazine'.

## Copyright



***Issue Theme: 4GLs and Application Generators***

# Changing Religions

*In some people's eyes assembler programming is a sin. Chris Hall believes it fulfils a crucial role in computing today.*

People no longer consider assembler as an alternative to the crop of so-called structured high-level languages such as C or Pascal. For years they have been brainwashed into (wrongly) believing assembler to be evil. There are applications for which assembler is ideal. Assembler should not be excluded on quasi-religious grounds. Don't be fooled by those who preach: 'Assembler Considered Harmful'. We should use the right tool for the right job. And if that means lapsing into assembler, then assembler it will have to be.

Assembler is at its best where speed and/or size are critical. However, if portability is a requirement, then (obviously) you cannot use assembler.

If you have tight memory or speed constraints, then assembler is what you need. If memory is short you might need to code entirely in assembler. If speed is your sole concern, then you can most likely identify key functions for hand-coding in assembler. This is the practice of fine-tuning that 10% of code which the program spend 90% of its time executing.

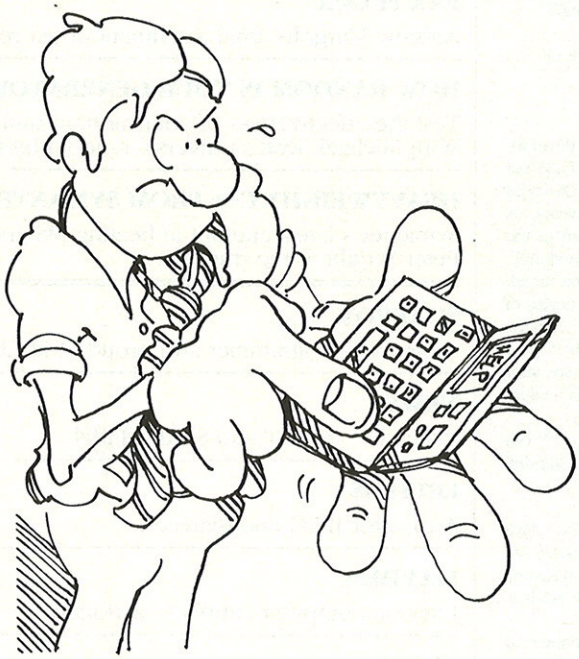
Compilers are getting better at generating code; processors are getting better at running the code they churn out. Still, humans are better code generators than compilers. With assembler, you get less code, so it runs faster. And it's smaller too.

Even on a machine overflowing with MIPS and megabytes, a dash of assembler can make all the difference, especially when dealing with the response time for the user-interface. Device drivers and the like are also excellent candidates for assembler. Given the choice wouldn't *you* buy the faster device with the smaller memory overhead?

Then there are the machines where build cost is critical. If you want to ship a $400 hand-held computer, then the difference between a $1 8086 (or even a 50¢ Z80) and a $20 386 is highly significant! By writing code in assembler, you can get the utmost from limited processor speed and memory. With embedded software, such as the software in printers, assembler can improve performance (eg a faster PostScript interpreter) or reduce the processor and/or ROM requirement, thus reducing the overall cost of the unit.

The benefits of using assembler must be balanced against the costs. Can the cost of, say, doubling the speed of a given program, be justified?

Opponents of assembler often claim that it is quicker to write programs in high-level languages, and the result is cheaper to test and maintain. The same could be said of 4GLs and application generators.



This is only partially true. In assembler, the programmer has to deal with more detail, implying more work. Assembler does not support or encourage any program structure - the programmer has to enforce this himself. Assembler code is not inherently readable, so the programmer must compensate by commenting liberally.

This does not mean, however, that there is any fundamental difference between assembler and high-level language programming. All the structured programming lessons we have learnt apply equally well to assembler as to C or Pascal. If there *is* a difference between assembler and high-level languages, then it is that greater attention must be paid to good practice with assembler, because it is an unforgiving language.

There is a dangerous notion that high-level language code can be largely 'self documenting'. The situation worsens in the domain of 4GLs and application generators. The danger is that programmers become lazy about commenting their code - three months later, something that once looked obvious now takes time to re-interpret.

Assembler does not add significantly to the design cost of a program. The testing and maintenance costs of a program are most influenced by the quality of the design and the level of understanding that can be gleaned from (commented) code.

Coding contributes a relatively small proportion of the total cost of a project. I believe the extra cost of using assembler, instead of C or Pascal, does not increase significantly the total cost of a project.

To write cost-effective assembler requires special skills and a disciplined approach. The general fear of assembler is self-fulfilling. If you don't have the necessary expertise, then you can indeed produce horribly unstructured, undocumented, unreliable and unmaintainable assembler. If you never use assembler because of the 'well known drawbacks', then you'll never develop the expertise.

Instead of using assembler, some developers will relax speed or space constraints, or simply upgrade processor or memory requirements. They may struggle to tune their high-level language code, to fool their compiler into yielding the required performance - all of which effort may be negated by the next compiler release.

Assembler is appropriate in special circumstances. In the right hands it is more cost effective than many think. It is said that an engineer is someone who can make for 10p what any fool can make for £1. Anyone who won't even consider assembler -- because it's a Bad Thing -- is not an engineer!

.EXE

---

*Chris Hall is the Managing Director of Locomotive Software, who has been perfecting assembler skills for more years than he cares to think about. He can be contacted by telephone on 0306 742140, or by email: chris@locomotive.com.*

# GREY MATTER

*Prigg Meadow, Ashburton*
*Devon TQ13 7DF*

Prices do not include VAT or other local taxes but do include delivery in mainland UK. Please check prices at time of order as ads are prepared some weeks before publication. This page lists some products – call us for a complete price list. ORDER BY PHONE WITH YOUR CREDIT CARD

**(0364) 654100**

FAX: (0364) 654200

## C & C++ COMPILERS

*NEW MICROSOFT VISUAL C++ 32-BIT EDITION is the compiler used by Microsoft to build Windows NT. Fully NT-hosted, with multi-threaded IDE, enhanced debugger, advanced Pentium optimisations, 32-bit version of MFC 2. It also targets and includes Win32s for those with Windows 3.1. It is only available on CD-ROM, which is fully exploited by Books Online - more than 10,000 pages of documentation, fully indexed and cross-referenced. Is this "Information At Your Fingertips"? Printed Docs available separately for those who read in bed! UPGRADE FROM VISUAL C++ PROFESSIONAL - **ONLY £74.** NEW PURCHASE - **ONLY £290.***

*NEW SYMANTEC C++ 6.0 PROFESSIONAL now has a Windows-hosted IDE, full AT&T 3.0 C++, targets 16 & 32-bit DOS, Windows & Win32s, royalty-free DOS extender, new Visual Programmer by Blue Sky promises to be more visual than Visual C++, MFC 2, full drag'n'drop, direct support for PVCS from IDE, Pentium optimisations, OPTLink/Windows, CD-ROM with online documentation, and lots more! Does it all work? Call us to find out. SPECIAL OFFER - **ONLY £149.** STANDARD EDITION **ONLY £85.***

*NEW WATCOM C32 FOR DOS must be the bargain of the year for those of you with BIG programs but are allergic to Windows. Royalty-free DOS extender with 32Mbyte Virtual Memory, same Pentium optimisations as the full C/C++32.* ***ONLY £135.***

| | |
|---|---|
| Aztec Embedded C86 | £290 |
| Borland C++ 3.1 | £250 |
| Borland C++ & AFX 3.1 | £365 |
| Microsoft Visual C++ Std | £95 |
| Microsoft Visual C++ Prof | £222 |
| MS QuickC 2.5 | £50 |
| MS QuickC & QuickAsm | £83 |
| TopSpeed C Prof | £180 |
| TopSpeed C/C++ Prof | £225 |
| Turbo C++ 3.0 | £59 |
| Turbo C++ Visual Ed. for Win | £84 |
| Watcom C/C++16 9.5 | £325 |
| Borland C++ for OS/2 | £250 |
| High C/C++ 386 3.1 | £555 |
| MS Visual C++ 32-bit Edition | £290 |
| NDP C/C++ 4.2 (386) | £445 |
| Symantec C++ 6.0 Prof | £149 |
| Symantec C++ 6.0 Std | £85 |
| Watcom C32 for DOS 9.5 | £135 |
| Watcom C/C++32 9.5 | £375 |

## LOW PRICES FOR MICROSOFT & BORLAND PRODUCTS

| | |
|---|---|
| **Microsoft Access 1.1** | **£275** |
| **Microsoft Delta** | **£222** |
| **Microsoft Fortran 5.1** | **£95** |
| **Microsoft Fortran Power Station** | **£222** |
| **Microsoft FoxPro 2.5 DOS** | **£280** |
| **Microsoft FoxPro 2.5 Win** | **£280** |
| **Microsoft Macro Assembler 6.1** | **£95** |
| **MS Multimedia Viewer Pub Kit** | **£225** |
| **MS Visual Basic for DOS Std** | **£95** |
| **MS Visual Basic for DOS Prof** | **£222** |
| **MS Visual Basic for Win Std** | **£95** |
| **MS Visual Basic for Win Prof** | **£222** |
| **Microsoft Visual C++ Std** | **£95** |
| **Microsoft Visual C++ Prof** | **£222** |
| **Microsoft Visual C++ 32-bit** | **£290** |
| **Microsoft Windows 3.1** | **£70** |
| **MS Windows SDK 3.1** | **£165** |
| **Borland C++ 3.1** | **£250** |
| **Borland C++ & AFX 3.1** | **£365** |
| **Borland Pascal with Objects 7.0** | **£250** |
| **dBase IV 2.0** | **£415** |
| **dBase Compiler** | **£250** |
| **Paradox Engine & DFX 3.0** | **£125** |
| **Turbo C++ 3.0** | **£59** |
| **Turbo C++ Visual Ed. for Win** | **£84** |
| **Turbo Pascal 7.0** | **£84** |
| **Turbo Pascal for Windows 1.5** | **£99** |
| **Paradox 4.0** | **£415** |
| **Paradox for Windows** | **£335** |

**WITH FULL TECHNICAL SUPPORT**

## DOS EXTENDERS

*NEW PHAR LAP TNT DOS-EXTENDER SDK 6.0 now lets users of Visual C++ 32-bit Edition build 32-bit DOS applications which take advantage of powerful Windows NT features such as DLL's, threads & multi-tasking. It also includes a 32-bit version of CodeView.* ***ONLY £340.***

| | |
|---|---|
| Phar Lap 286IDOS-Ext SDK | £330 |
| Phar Lap TNT DOS-Ext SDK | £340 |

## EDITORS

| | |
|---|---|
| Borland Brief 3.1 | £125 |
| Codewright 2.0 for Windows | £205 |
| Codewright NT | £205 |
| EditPro for Win | £200 |
| Kedit 5.0 | £125 |
| Multi-Edit Standard 6.1 | £115 |
| Multi-Edit Professional 6.1 | £165 |
| Norton Editor 2.0 | £55 |
| QEdit Advanced | £45 |
| Vedit-Plus 3.5 | £140 |

## C & C++ FOR DOS

### Comms

| | |
|---|---|
| Asynch Prof for C/C++ | £130 |
| C Asynch Manager | £145 |
| C Comms Toolkit | £215 |
| COMM-DRV | £145 |
| Essential Comms | £215 |
| Greenleaf CommLib 4.0 | £225 |
| SilverComm "C" Asynch 4.0 | £135 |

### Database

| | |
|---|---|
| Btree Filer for C Network | £170 |
| Btrieve | £365 |
| C-Index+ 4.0 | £250 |
| c-tree Plus 6.0C | £565 |
| CodeBase 5.0 | £240 |
| D-ISAM | £325 |
| d-tree | £490 |
| DataBoss | £520 |
| Greenleaf Database Library | £180 |
| Paradox Engine & DFX 3.0 | £125 |
| POET SDK 2.0 | £360 |
| r-tree | £295 |
| Raima Database Manager | £305 |
| Raima Object Manager | £305 |
| SoftFocus Btree/ISAM | £75 |
| Sycero.C Net | £895 |
| Topaz for C 4.0 | £215 |
| Watcom SQL Developer's | £260 |

### Graphics & GUI

| | |
|---|---|
| Accusoft Image Format Lib | £360 |
| dGE 5.0 | £205 |
| Essential Graphics Chart | £130 |
| Essential Graphics Kernel | £135 |
| Essential Graphics GUI | £99 |
| GraphiC 6.1 | £305 |
| graphics-MENU & Data Entry | £205 |
| GX Graphics 2.0 | £155 |
| HALO Professional 2.0 | £245 |
| MetaWINDOW-DOS 4.3 | £150 |
| MetaWINDOW-DOS/286 4.3 | £210 |
| MetaWINDOW-DOS/386 4.3 | £270 |
| object-Menu | £305 |
| PCX Toolkit 5.0 | £155 |
| PCXLab-DOS/286/386 | £150 |
| Real-Time Graphics & M/CT | £185 |
| T-Windows | £240 |
| TEGL Windows Toolkit 3.0 | £180 |
| zApp for DOS Graphics 2.0 | £330 |
| Zinc 3.5 for DOS | £354 |

### Math & Scientific

| | |
|---|---|
| C/Math Toolchest & Grafix | £45 |
| Huge Virtual Array & NAT | £215 |
| Linpack.h++ w/Source | £429 |
| Math.h++ w/Source | £257 |
| MIRACL | £39 |
| Science & Eng Tools 8.0 | £115 |

### Printer

| | |
|---|---|
| Baby Driver Kernel | £185 |
| BGI Printer Driver Toolkit | £105 |
| GX Printer | £240 |
| SLATE | £260 |
| TesSeRact Printer Control Sys | £45 |

### Screen

| | |
|---|---|
| C-Scape 4.0 | £360 |
| C/Windows Toolchest | £45 |
| Greenleaf Datawindows 3.0 | £225 |
| MEWEL Window System 4.0 | £380 |
| Object Professional for C++ | £165 |
| Panel Plus II | £260 |
| TCXL-DOS 6.1 | £79 |
| Turbo Vision Dev Toolkit 2.0 | £105 |
| Vermont Views Plus | £595 |
| zApp for DOS Text | £330 |

### General & Systems Libraries

| | |
|---|---|
| C Tools Plus/6.0 | £115 |
| CAD/CAM Dev Kit/2D Personal | £375 |
| CodeRunneR | £150 |
| Heap Expander 3.0 | £130 |
| Hold Everything | £130 |
| Interwork | £205 |
| MoreHeap | £110 |
| MTASK | £215 |
| Multi-C w/Source | £39 |
| PDK #1, #2, #3 | £95 |
| PowerSTOR | £99 |
| Resident-C | £160 |
| TesSeRact Compression Sys | £45 |
| TesSeRact TSR Dev Sys 2.0 | £79 |
| TG Professional (CAD/CAM) | £395 |
| Tools.h++ w/Source | £257 |
| TSRs and More | £115 |
| VMData | £370 |

### Tools

| | |
|---|---|
| C-DOC 5.0 | £150 |
| C-Vision 3.1 | £90 |
| Clear+ for C | £135 |
| CodeCheck Personal | £250 |
| MKS LEX & YACC | £240 |
| PC-Lint | £85 |
| PCYACC Personal | £250 |
| SourcePrint+ Deluxe 5.0 | £190 |

## PROGRAMMING tools

| | |
|---|---|
| **Ada** | **Assemblers** |
| **Basic** | **C/C++** |
| **Comms** | **Cross Dev** |
| **Custom Controls** | **Database** |
| **Debuggers** | **Dos Extenders** |
| **Editors** | **Fortran** |
| **Graphics** | **GUI** |
| **Linkers/Locaters** | **Lisp** |
| **Modula-2** | **Multi-tasking** |
| **Pascal** | **Prolog** |
| **Rexx** | **Scheme** |
| **Smalltalk** | **SQL** |
| **Version Control** | **Visual Programming** |
| **Windows** | **Xbase** |

***We stock many items for which there is no space in these advertisements.***

## VISUAL BASIC FOR WINDOWS

| | |
|---|---|
| Visual Basic 3.0 for Win Std | £95 |
| Visual Basic 3.0 for Win Prof | £222 |
| Comms Lib/Win | £115 |
| Network Library/Win | £80 |
| PDQ Comm for Windows | £99 |
| Agility/VB | £160 |
| Btrieve for Windows | £365 |
| CodeBasic | £145 |
| ObjecTrieve Prof 2.01 | £410 |
| Q+E Database/VB | £99 |
| Q+E Multilink.VB | £295 |
| Q+E ODBC Pack | £140 |
| Raima Data Manager | £380 |
| VB/ISAM MU | £150 |
| VBtrv Control Pack | £140 |
| Visual Controls for Btrieve | £180 |
| Visual/db Network 3.0 | £235 |
| 3-D Gizmos | £80 |
| ButtonTool & EditTool | £69 |
| ChartBuilder for VB | £75 |
| Control Palette 2.0 | £110 |
| Custom Control Factory | £49 |
| MS Visual Control Pack | £69 |
| QuickPak Pro for Win 3.0 | £145 |
| Spread/VBX | £205 |
| SpyWorks-VB | £120 |
| VB/Magic Controls | £120 |
| VBAssist 3.0 | £135 |
| VBTools/Win 3.0 | £145 |
| 3D Graphic Tools | £130 |
| Compression Plus for VBWin | £210 |
| GUI Assist | £350 |
| ImageKnife/VBX | £145 |
| Microhelp Muscle/Win | £160 |
| PCX Toolkit for Windows | £155 |
| QuickPak Scientific for Win | £105 |
| Spellcheck/VB | £49 |
| TrueGrid | £80 |
| VB Compress Pro | £95 |
| VBXRef Utility | £80 |

## VISUAL BASIC FOR DOS

| | |
|---|---|
| MS Visual Basic for DOS Std | £95 |
| MS Visual Basic for DOS Prof | £222 |
| Comms Lib/DOS | £115 |
| Network Library/DOS | £80 |
| PDQ Comm | £65 |
| Btrieve 5.1 | £365 |
| db/Lib Prof 3.0 | £195 |
| E-Tree Plus | £165 |
| Elltech's GUI Toolkit | £125 |
| Graphics QuickScreen | £105 |
| Graphics Workshop | £105 |
| GraphPak Pro | £99 |
| GX Graphics 2.0 | £155 |
| Compression Plus | £130 |
| Compression Workshop | £110 |
| Microhelp Muscle/DOS | £155 |
| PDQ | £110 |
| Printer Plus | £185 |
| ProBas 5.5 | £165 |
| QuickPak Prof 4.1 | £145 |
| QuickPak Scientific 3.0 | £105 |
| QuickScreen 4.0 | £105 |
| Science & Eng Tools 8.0 | £80 |
| TSRific | £80 |

## BASIC

### Compilers & Interpreters

| | |
|---|---|
| BBC BASIC-86 Plus | £75 |
| CA-Realizer (Win) | £190 |
| GFA-BASIC for DOS | £85 |
| GFA-BASIC for Windows | £170 |
| GFA-BASIC for Win Compiler | £42 |
| PowerBASIC 3.0 | £110 |
| TrueBASIC 3.0 | £75 |

## News & Views

Windows NT is Now There and so is Microsoft's Visual C++ 32-bit Edition. Apart from being the same compiler which Dave Cutler uses, it is the first developer's tool which exploits the power of CD-ROM and ships without any printed documentation (should save a few trees!). It includes a cut down version of Phar Lap's new TNT DOS Extender, which lets you develop 32-bit DOS apps which incorporate threads & DLL's. Who needs NT?! Once you have tried it, call us to buy a full copy.

For those of you Bored with Borland and Miffed with Microsoft, why not check out the new Symantec C++ 6.0. Preliminary reports suggest that it will be more visual than Visual C++ with faster compilation and linking, and some novel features in its' IDE. The big question is - WILL IT WORK? If it does, and the documentation is adequate, it could supercede both Borland C++ and Visual C++ as the most productive C++ compiler for Windows. Call us to find out more about it.

## C & C++ FOR WINDOWS

| | |
|---|---|
| C Comms Toolkit/Windows | £270 |
| COMM-DRV | £145 |
| CrystalComm for Windows | £130 |
| Greenleaf Comm++ | £180 |
| Greenleaf CommLib Pro | £355 |
| Agility/VC++ | £160 |
| Btrieve for Windows | £365 |
| CB/ISAM MU for Win | £150 |
| CodeBase 5.0 | £240 |
| Integra VDB | £399 |
| Paradox Engine & DFX 3.0 | £125 |
| POET SDK 2.0 | £360 |
| Q+E Database Library | £295 |
| Raima Database Manager | £380 |
| Raima Object Manager | £305 |
| Watcom SQL for Windows | £260 |
| Essential Graphics Chart | £300 |
| GraphiC/Win | £370 |
| Graphics Server SDK | £205 |
| Windows Charting Tools | £300 |
| Accusoft Image Format Lib | £360 |
| Control Palette 2.0 | £110 |
| Data Entry Workshop | £130 |
| DataTable | £200 |
| Diamond Toolkit | £310 |
| Magic Fields | £255 |
| MS Visual Control Pack | £69 |
| Out of Controls w/Source | £355 |
| PCX Toolkit for Windows | £155 |
| ProtoGen+ | £295 |
| Toolbox for Windows 3.0 | £305 |
| Win/Sys Library | £105 |
| WindowsMaker Pro 4.0 | £605 |
| zApp for Windows 2.0 | £330 |
| zApp for Windows NT 2.0 | £330 |
| Zinc 3.5 for Windows | £354 |
| C++ Designer | £195 |

**Oracle class lib**

*Rogue Wave's DB.h++ database class library for C++ has been updated to support Oracle V7.0 and Visual C++. The company is targeting this release at users of Oracle who are planning to move to C++. DB.h++ V1.5 is available on several platforms including DOS, Windows 3.x, OS/2 and a number of UNIX platforms. The DOS version is priced at £430 (£599 for UNIX). It is distributed in the UK by Hypersoft (0273 834596).*

**New R&R**

*The xBASE edition of R&R Report Writer for Windows (RRW) is now in its second incarnation. The new version addresses a number of the limitations of its predecessor including the lack of a vertical scroll bar, inability to embed bitmaps into reports and no mouse-driven line/box drawing. It additionally provides several new predefined functions and page preview options. There's also improved parameter passing in the run-time version. RRW V2.0 costs £199 for a single user. Upgrading from RRW V1.0 will cost £35. R&R is on 0628 788181.*

**Eiffel on SPARC**

*Austin-based Tower Technology will be launching its TowerEiffel programming environment at OOPSLA '93 which is being held in Washington DC between September 26th and October 1st 1993. TowerEiffel is an implementation of Eiffel for Sun SPARC platforms which provides a 'tight' C interface, smart recompilation and support for team development projects, distributed applications and commercial library development. The US price is $1,295 for a single commercial licence ($345 for personal use). Tower technology is on 0101 512 3286406.*

**TestCenter**

*TestCenter is a new UNIX C/C++ development tool from CenterLine for Sun workstations for testing code. It features automatic run-time error checking, memory leak detection on executables and GUI-based batch and regression testing. A scripting language is provided for querying results of multiple test runs. Applications can be tested without recompilation. K2 Software Development (061 7778118) is offering TestCenter at an introductory price of £2,500.*

## SMS on Windows

SMS, the version control system from IntaSoft is now available under Windows. SMS V4.0 provides both DOS and Windows support 'out-of-the-box'. It comprises tools for version control, configuration management, change management and categorised modification requests. The version control component allows developers to display revision history by name, revision number, creation date, author and purpose. Modification requests are used to assign tasks and track their progress using email. And configuration management offers a mechanism for grouping files and splitting development paths. SMS also provides user-defined report formatting, dependency analysis, a language-independent macro preprocessor and a dependency generator. The single-user version is priced at £490 (5-user is £1180). For a limited time IntaSoft is offering an upgrade from the single-user DOS package for £100 (£200 for multi-user). IntaSoft is on 0392 217670.

## Install-o-gram

InstallShield V2.0 is the latest version of the install program development kit from Illinois-based Stirling Technologies. Unlike some installation toolkits available which are based on C libraries, InstallShield doesn't require a knowledge of C. Instead, Stirling has provided its own scripting language. This has been made more comprehensive in V2.0, with the addition of new variable types (Long, Short, List, Pointers, Chars, Structures, Typedefs); local/global variables; enhanced list & array handling functions; procedure/function calls and switch statements and string assignment and cocatenation functions. In addition, developers can now include multiple script files. Stirling claims V2.0 is significantly faster than its predecessor.

There is now support for Windows 3.1, with functions to add, delete, enumerate and modify ProgMan groups and items. Stirling has also included version-checking and functions for maintaining registration databases. Other features include improved file compression/decompression, support for calling DLLs, customisable status bar and, according to Stirling, 'more intelligent, safe updating' of AUTOEXEC.BAT/CONFIG.SYS files.

The UK distributor is Systemstar SoftTools (0992 500919). InstallSheild is priced at £295.

## 32-Bit Visual C++

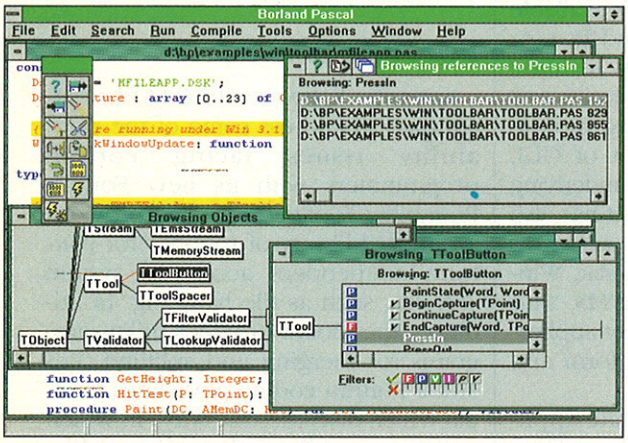
Microsoft's new NT-hosted C++ compiler, Visual C++ 32-bit Edition (VC4NT), is now available. The company has thoughtfully decided to do away with the impressive stack of diskettes from Visual C++ installations; instead it has opted for the single CD-ROM favoured by SunSoft. Printed docs have gone as well: all 9000 pages worth are on the CD-ROM. As a result, the complete, boxed VC4NT is a featherweight at 500 grams. However it *is* a big package. Remember when you first installed VC++, and ran CHKDSK: 50 MB gone. Well VC4NT is even larger, needing at least 80 MB for a complete installation; although this can be cut down to 6 MB by running the compiler from the CD-ROM.

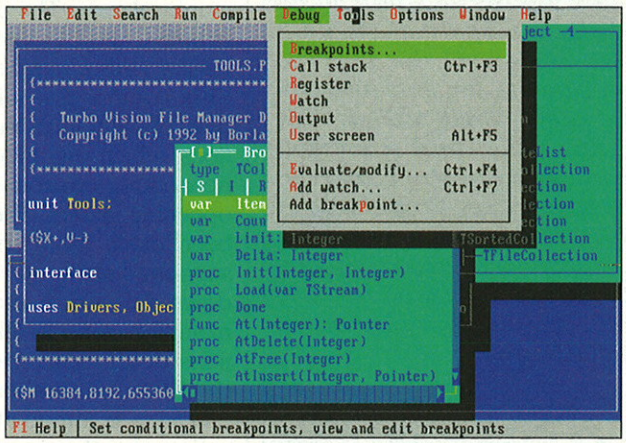
VC4NT has a similar suite of tools to VC++. So there are 32-bit versions of MFC V2.0, AppWizard, Class Wizard, AppStudio and Visual Workbench. A number of enhancement have been made to the debugger for debugging NT applications. These include multiple thread support, structured exception handling and a memory window. In addition, Microsoft has included Spy++ with VC4NT, a new tool which displays information on threads, processes and windows in Win32-based applications. Another improvement to VC++ is a grep-like find-in-files facility for performing global searches in multiple source files. Microsoft has also integrated the profiler into the VC4NT Visual Workbench.

Microsoft ships supports for both Win32 (ie NT) and Win32s (Windows 3.1) targets out of the same box. Until the end of October, existing users of VC++ Professional can upgrade for £69. Printed documentation is available separately in two kits priced at £69 each. The first contains the *User Guides* and *Language References*; the second provides *Win32* and *Win32s* references. Details are available from the Microsoft Upgrade Centre (081 6148000).

# The world's fastest and most widely used language

Autumn Catalogue OUT NOW!




## Borland Pascal 7.0

Borland's Pascal products set the standard for the Pascal language. And with good reason. They're faster, easier to learn and take you further in professional programming. With the power of Pascal at your command there are no limits.

Borland® offers the Pascal to suit you:

| | DOS | Windows |
|---|---|---|
| Professional | Borland Pascal with Objects 7.0 | |
| Entry Level | Turbo Pascal 7.0 for DOS | Turbo Pascal for Windows 1.5 |

### Borland Pascal with Objects™ 7.0

The highest productivity professional programming system ever! Borland Pascal with Objects™ 7.0 has everything the professional developer needs for DOS, DPMI and Windows development in a single box. And the built-in Application Frameworks™ give you great-looking user interfaces in just a few lines of code. You'll be amazed with the compiler's speed. There is no better professional development system.

OBJECT COMPUTING NOW!

**Competitive upgrade price from any BASIC development tool £129.95 + VAT.**

**Recommended Retail Price. £299.95 + VAT.**

### Turbo Pascal® for Windows 1.5

Turbo Pascal for Windows 1.5 makes it easy to program in Windows. An on-line help system with over 100 sample programs helps with application development. And the enhanced Windows IDE includes colour syntax highlighting, SpeedBar,™ and Turbo Debugger® for fast and easy debugging. Turbo Pascal for Windows gives you full support for features including Multimedia and Object Linking and Embedding (OLE). You can visually create objects with the Resource Workshop.™

No matter what your development experience, Turbo Pascal for Windows is the easiest way to learn Windows programming.

**Recommended Retail Price. £119.95 + VAT.**

## Turbo Pascal 7.0

### Turbo Pascal® 7.0 for DOS

Turbo Pascal 7.0 is the ideal way to take advantage of the ease of object oriented programming to create DOS real mode programs. The many powerful features include a fast compiler and high-capacity DOS IDE with colour syntax highlighting to make your code easy to read and correct, and unlimited undo/redo. Object Browser™ lets you navigate through your source code with ease, and the on-line hypertext help system provides over 1Mb of help text and copy-and-paste examples. Turbo Pascal 7.0 really is the easiest way to learn object oriented programming for DOS.

**Recommended Retail Price. £99.95 + VAT.**

### Leader in Object Computing

Borland is the leading supplier of Pascal solutions with more than two million sold. Borland's compilers and tools are carefully crafted to meet all your needs - building on generations of Pascal products. Borland will deliver products that meet your future requirements by continuing to supply award winning Pascal solutions for DOS, DPMI and Windows.

## See your dealer today

For further information, or details of your local dealer call

**0800 212727**

or complete the coupon.

**Please send me the FREE Pascal information pack.** Ref: AS/9339/BQ

Name_____

Position _____

Company_____

Address_____

_____

Postcode _____ Tel. _____

**Please return to: Borland International (UK) Ltd, FREEPOST, PO Box 9, Stroud, Glos., GL6 8BR.**

# Borland

Power made easy

➤ CIRCLE NO. 062

Borland International (UK) Ltd, 8 Pavilions, Ruscombe Business Park, Twyford, Berkshire RG10 9NN Tel 0734 321150

**Brillo Brillig**

*Decompile your Clipper 5.x .EXE and .OBJ files using Brillig from APTware. Clipper p-code is converted back into source, and .RMK and .LNK files are generated for rebuilding the application with RMAKE. Along with the ability to extract variable, function and procedure names, Brillig will also format the code and comment procedure and function headers in user-supplied styles. Linker support includes Blinker, RTLink and MS-link. APTware (0491 826060) is offering Brillig at a special introductory price of £169 (usual price £199) until the end of November.*

**SQL from Clipper**

*Strike! for DOS is a Clipper add-on which provides Clipper 5.x developers with a way to connect to Gupta SQLBase, Microsoft SQL Server and Oracle using a single API. It costs £395 and is available form QBS Software (081 9944842) and Applied Network Solutions (0276 452064).*

**NT Graphics Server**

*Bits Per Second (0273 727119) has introduced an NT version of Graphics Server, the company's graphing and charting DLL. The complete package includes both the NT and Window 3.1 versions of Graphics Server. It is priced at £395. Existing users can upgrade from Graphics Server V2.x for £95.*

**MKS for OS/2 V2.x**

*Use UNIX utilities on OS/2 V2.x with V4.2 of the MKS Toolkit. This release offers several new utilities including:* `awkc`, *an* `awk` *compiler for developing standalone* `awk` *programs;* `make` *and* `iconv`, *for converting from one code set to another. There are 130 in total, all of which have been rewritten for 32-bits. In addition, MKS V4.2 is POSIX-compliant. Grey Matter (0364 654100) is selling MKS V4.2 for £230.*

**PC-to-Video**

*Transfer text, graphics and animations from a PC to a VCR/TV with TV Coder, WestPoint Creative's latest video output card. It features a built-in flicker filter and a programmable RGB colour lookup table which can be used for enhancing the picture or generating special effects. TV Coder costs £159. Phone WestPoint Creative on 0734 248590 for details.*

## New Galaxy

Visix Software has announced release 2 of its Galaxy Application Environment, a cross-platform development environment for creating large-scale, distributed graphical apps. The product is available in both C and C++ versions, and the new release adds a number of new features. Visix comprises a suite of libraries and development tools for building applications which are independent of GUI, networking environment or underlying operating system. The product has also been made more consistent across its supported platforms of UNIX, Mac, Windows, NT, OS/2 and OpenVMS, thus making it easier to take Galaxy applications developed on one platform and run them on another.

Galaxy is aimed at corporate application developers and independent software vendors who build high-end commercial applications and who wish to be able to support multiple platforms, or who suspect/fear that the platform for which they are currently developing will soon be superseded. Visix can be contacted on 071 8725825. Starting price is £6,700 per developer. There is no run-time licence.

## Portable Fortran

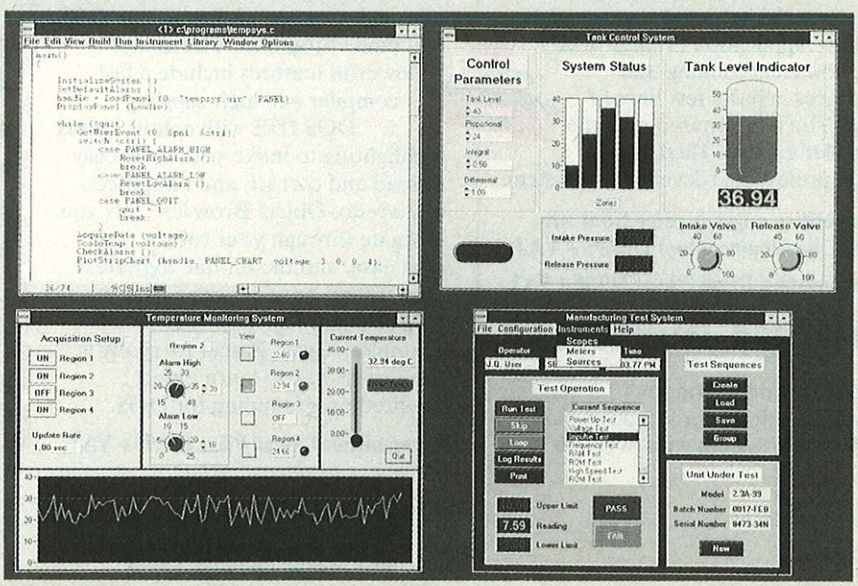
Scientific Software Ltd (SSL) has attempted to address some of the portability issues facing Fortran programmers with its new Fortran Portability Toolkit (FPT). This provides a standard library of routines for platform-independent access to system services, such as file handling. In addition, the toolkit provides utility programs for merging and splitting files and a Fortran coding guidelines manual which discusses the ways in which a program can be made more portable. FPT is available on UNIX and VMS workstations for the introductory price of £450 until the end of November. SSL is on 0491 411727.

## Instrumentation Software

National Instruments has launched LabWindows/CVI (C for Virtual Instrumentation), an instrumentation development package that uses ANSI C under Windows and SPARCstations. The software is an automatic code generator designed for engineers and scientists who want to use traditional programming tools for developing instrumentation systems. The development environment includes built-in libraries for performing data acquisition, analysis and presentation, as well as a user interface editor. The system also simplifies the setup and control of the external devices such as GPIB, VXI and RS-232, by providing over 300 ready-to-use drivers for specific instruments.

The Windows version also controls the National Instruments plug-in data acquisition and SCXI signal conditioning hardware.

The analysis library contains over 200 functions for signal processing, filtering, windowing, statistics, curve fitting, interpolation and matrix algebra. Interprocess communication is supported with TCP/IP on the PC and Sun, and via DDE under Windows on a PC. National Instruments is on 0635 523545.



# SoftTools for the Job

## NEW! NEW! NEW! NEW!

### MEWEL 4.0

**Single Source - Multiple Environments**

New release of the **MEWEL** user interface library from Magma Software Systems. A library which allows the developer to write a single set of source code for Windows, DOS Gaphics, DOS Text, OS/2 and UNIX. MEWEL is an implementation of the Windows API.

**MEWEL** is particularly recommended for zApp 2.0 and other C++ framework users and for expanding class libraries supplied with compilers.

**Price: MEWEL/Text for DOS - £365 - with full source £695**
**MEWEL/GUI for DOS - £365 - with full source £695**

## NEW! NEW!

### InstallSHIELD v2.0

**Latest Release of Installation Program for Graphical Applications.**

**InstallSHIELD** has become the standard tool for building bullet proof Windows and OS/2 installation programs. A new 1000 page manual describes the many new advanced features including multi-language support. Existing users may upgrade - call for details.

**Price: Windows £265 OS/2 £520**

### MORE SHIELD Development Products

**TbxSHIELD**
**Windows £265 OS/2 £265**
A toolbox library that allows sophisticated tool box controls to be created quickly and easily.

**LogSHIELD**
**Windows £340 OS/2 £520**
A graphical session recording and playback facility.

**MemSHIELD**
**Windows £340 OS/2 £520**
High performance memory management library.

**DbxSHIELD**
**Windows £340 OS/2 £520**
System for implementing the logic behind dialog boxes in an application.

**ResourceSHIELD**
**Windows £155 OS/2 £355**
A fully integrated editing system for resource files.

### Phar Lap 286|DOS Extender

***now with support for Microsoft Visual C++ and includes 286|VMM at no extra charge***

With the **286|DOS Extender** and your Microsoft C, Borland C++ or MS Fortran and **now Microsoft Visual C++** compiler, you have all the tools necessary to quickly build protected mode applications - often by relinking without making source code changes. Protected mode applications can be built that access up to 16 megabytes of memory on any DOS 80286, 386, 386SX or i486 PC. **286|DOS Extender** is also compatible with Borland's Turbo Debugger and Microsoft's linker and CodeView debugger. **Includes 286|VMM to access Virtual Memory too!**

**Price: £375**

### KnowledgeMan

***NEW - Release 3.1 from mdbs***

The latest release of the **KnowledgeMan** fully integrated professional rdbms moves even closer to an event driven GUI environment. Enhancements include Compiler and User-Defined Functions, increased performance, improved system command interface, Optimized Query by Example, new Report Definition Language and Binary Large Objects.

**KnowledgeMan** capabilities have always gone beyond those of the competition. **Version 3.1 features and enhancements make it even easier to develop applications rapidly and without compromise.**

**KnowledgeMan** DOS SU - £990, OS/2 SU - £1320
**KnowledgeMan** DOS LAN - £1990, OS/2 LAN - £2320
Call for update and SUN or VAX prices.

## NEW! NEW!

### QUICTURE

**Compress Graphics in Word for Windows**

**Quicture** is a unique graphics handling utility for Word for Windows. By using Quicture graphics are compressed and off-loaded to disk replacing each one with a placeholder. You can view and print graphics at any time but in draft mode, documents scroll, save and print quickly, taking up less space. From WexTech, the authors of Doc-To-Help.

**Price: £75**

## NEW! NEW!

### ISCL

**The largest library of C++ Shareware in the World?**

Claimed by Image/ISoft to be the largest library of public domain and shareware C++ source code in the world. **ISCL** includes the GNU 386 C++ compiler, a comms library, maths and matrix classes, DPMI programming kit, classes for Btreive and Paradox and a C++ library for building GUIs. **ISCL** contains over 100 MBytes of code and is available only on CD-ROM.

**Price: £95**

### Object/Designer

***Create Windows applications as if your job depends on it!***

Out performing Visual C++, WindowsMaker and Case W, **Object/Designer** enables developers to quickly generate all application source files needed for a Windows application. Includes support for zApp 2.0, Microsoft C++/MFC, Borland C++/Object Windows, Turbo Pascal for Windows and CommonView with instant access to third party editors. The features of **Object/Designer** from Image/Soft make it the only code generation tool flexible enough to be used to write full function Windows applications

**Price: £370**

### SoftTools from TeraTech

| | |
|---|---|
| **For C++** | |
| **Ace Comparator** | **£90** |
| **NOW!** | **£40** |
| **E-MEM** | **£90** |
| **Virt-Win** | **£235** |
| **For Visual Basic** | |
| **Dazzle/VB** | **£225** |
| **Dazzle/VB Professional** | **£375** |
| **FinLib/VB** | **£125** |
| **SpellCheck/VB** | **£55** |
| **ProMath** | **£105** |
| **VB/Lite** | **£90** |
| **Telecomm ToolKit** | **£75** |

**Other Products Supplied & Supported by Systemstar SoftTools:**

| | |
|---|---|
| **zApp** v 2.0 Win,DOS Text or Graphics | £330 |
| **zApp** v 2.0 Windows NT | £330 |
| **zApp** v 2.0 OS/2 | £430 |
| **C-scape** DOS £390, 386 | £790 |
| **Doc-To-Help** | £265 |
| **Dolphin C Toolkit** | £99 |
| **Dolphin Far Memory Manager** | £99 |
| **Phar Lap 386\|DOS-Extender** | £390 |

Please check prices at time of order.
Prices do not include VAT or carriage.
All Trademarks Acknowledged.

### Magna*Charter* II

***Build flowcharts in minutes!***

**Magna*Charter* II** for Windows lets you build any kind of chart in minutes and includes all standard flowcharting symbols which are displayed as icons or add your own. Use the crowsfeet for database schemas. Features are accessed by drop-down menus, dialog boxes and multiple windows with a wide choice of text styles and sizes. Editing is simple 'cut-and-paste'. A wide range of printers is supported as is PostScript.

**Price: £190**

STAD4

***For more information, demonstration disks or details of our 30 day trial offer call: (0992) 500919***

**SYSTEMSTAR**
**SoftTools Limited**

*1-3 Parliament Square Hertford SG14 1EX*
*Telephone: (0992) 500919 Facsimile: (0992) 554261*

**dBASE in Accounts**

*Stanford Software has developed Epsilon, an accounts package which reads and writes standard dBASE .DBF files. An API is available which documents the database fields and indexes used by Epsilon. Stanford suggests that this information can be used by xBASE developers to manipulate accounts from within tools such as FoxPro or Clipper. It includes an example Clipper program. Epsilon is priced at £349. Stanford is on 061 4804051.*

**VBW Shareware**

*VBasic Library V6.0 is now available from EMS. It is distributed on both diskette or CD-ROM and contains over 425 shareware and public domain Visual Basic programs and utilities. VBasic Library V6.0 costs $59.50 from EMS (0101 301 9243594).*

**LPA's VBW interface**

*LPA 386-Prolog V2.5 now provides direct access to the Windows GDI and a new configurable DDE toolkit for interfacing to Visual Basic. The Programmer Edition costs £745. It includes built-in predicates, development environment, debugger and a DLL interface written in C, C++ or Pascal. The Developer Edition (£1,495) additionally provides a run-time generator. LPA is on 081 8712016.*

**Superbase price cut**

*SPC (0344 867100) has cut the price of Superbase V2.0 for Windows. The single-user version is down from £545 to £299. The price of the Developer Edition has also dropped from £950 to £665.*

**Speaking Paradox**

*The Paradox Users Group and Richplum Ltd have organised the first European Paradox Conference. It is being held at London's Park Lane Hotel on 22nd and 23rd November 1993. It is aimed at users, developers and supporters of Paradox. Sponsors include Borland, WordPerfect, Dell and Paradox Informant. There will be 30 international speakers and sessions on Paradox for DOS/Windows (both user and programming aspects) and client/server issues. Registration is £495 (15% discount if booked before 15th October). Phone the conference office on 081 3327997 for details.*

## Add GUI to DOS Apps

MicroMini has introduced V1.4 of GUI Assist, a tool which works in conjunction with VBW to add a Windows-hosted front-end to text-based DOS applications, without requiring access to the original source code. GUI Assist provides a DLL which routes output from the DOS application to VBW forms and traps keyboard input. The Windows-hosted front-end is created using a new utility called GUI Gen. According to the manufacturer, this can generate VBW forms for the front-end automatically, achieved by running the DOS application and recording screen positions/attributes for the various program controls. These are then translated into equivalent VBW controls such as forms, command buttons, text/list boxes etc, along with `Click` event handlers. The developer is able to modify the user-interface using VBW.

There are two versions of the product. The Professional Edition (£1,500) includes the ability to control multiple DOS applications and comes with an install program for distributing GUI Assist applications A base version is also available at £350. MicroMini is on 0844 275666.

## Insider

A £25 donation to Amnesty International will secure you a copy of Codehigh's Inside Information which normally sells for £95. Inside Information is a PC-based directory of over 6000 PC products (software and hardware) from 2000 manufacturers. Information listed includes manufacturer, telephone number, RRP along with references to reviews/product announcements in the computer press (including a three-year index to *.EXE*). To receive your copy, send a cheque, payable to Amnesty International, Basingstoke Group, along with a note of your address and the size of diskette you prefer, to Codehigh Ltd, Sedgewell House, Sedgewell Road, Reading RG4 9TA. This offer ends at the end of December.

## Blinkety Blink

Blinker V3.0 from Blink Inc provides a Windows linker, a royalty-free 286 DOS extender, a dynamic overlay linker and memory swap function, all from the same package. Features of Blinker include: the facility to link Clipper 5 programs incrementally; up to 16 MB addressable memory and the ability to create applications which will run transparently in protected or real mode. It also includes a compiler-independent virtual memory system which gives up to 64 MB of virtual memory. For Clipper programmers it offers symbol table compression, a profiler and 'burning-in' of environment variables. Blinker V3.0 costs £229. Upgrades from previous releases are available (eg £75 from Blinker 2.x). It is distributed in the UK by QBS Software (081 9944842).

## DCE kit for SCO

The DCE Developer's Kit V1.0 from SCO is a suite of tools for developing distributed applications under SCO Open Server and SCO Open Desktop. The kit is an implementation of Release 1.0.2 of DCE from OSF. It provides an application development environment with an Interface Definition Environment (IDL) compiler and header files and run-time libraries to build DCE applications. DCE executives and servers are available for creating all components of DCE including: threads, remote procedure calls, security, cell directory services and distributed time services. The DCE Developer's Kit V1.0 is priced at £4,270 on diskette and £4,100 on tape. SCO is on 0923 816344.

## ODBC drivers kit

Now that ODBC has established itself as the Windows API for accessing databases, we should soon begin to see more and more ODBC-aware applications. ODBC drivers for popular databases are available from several vendors. Contemporary Software's new Q+E ODBC Driver Pack provides drivers for over 20 different file formats including Oracle, SQLBase, dBASE, NetWare SQL, SQL Server, Sybase, Informix, DBM, Paradox and XDB. The drivers support all Core and Level 1 ODBC functions, plus a number of Level 2 functions as well. Contemporary says its drivers will work in any ODBC-compliant application including Access V1.1, VBW V3.0 and Lotus 1-2-3 for Windows. The Q+E ODBC Driver Pack is priced at £149.

# MICROSOFT

## Training in Microsoft Tools

## VISUAL BASIC

Our Visual Basic Programming Course has been fully revised for Version 3.0.

It is a hands-on course with many practical exercises and a supporting case-study which illustrate the practical application of Visual Basic features and functions.

The course demonstrates:

- How Visual Basic can and should be used
- The strengths and weaknesses of Visual Basic
- How add-ins and other tools support the Visual Basic Programmer

This is a five day course. The first day is optional for those who already have experience of Windows programming.

THE course for those new to Visual Basic.

## VISUAL C++

Our 5 day Visual C++ programming course is for developers with a good, current knowledge of C. It provides a thorough and practical introduction to the Visual C++ programming environment, with the emphasis on writing commercial applications.

After the course delegates will:

- understand C++ extensions to C
- understand how principles of Object Orientation are implemented in C++
- be effective in the use of C++

All our courses are developed and led by people who are actively involved in the commercial application of software tools. This guarantees that they will understand your needs and the best way to meet your training requirement.

## WINDOWS

This introduction to Windows programming covers the most relevant areas in sufficient detail to allow programmers quickly to become efficient in producing applications. By the end of the course students will:

- understand Windows architecture, philosophy and application design considerations
- understand the appropriate use of DDE and DLL's
- be able to make effective use of the available features and functions.

This 5-day course is an invaluable introduction to Windows programming for competent C programmers who already have an appreciation of the Windows graphical interface.

HANDS-ON, PRACTICAL TRAINING IN SMALL CLASSES.

## ADVANCED WINDOWS

- Learn how to use advanced facilities; OLE, MDI, MAPI
- Explore Windows operating system variants and benefits; particularly WIN32 and WINDOWS NT
- Maximise programming efficiency

If this is what you need to do - we have the course for you.

Delegates need to be competent C programmers with at least three months experience of programming Windows applications, preferably in C/C++. Practicals involve coding in C++.

IF YOU NEED ADVANCED TRAINING....CALL FOR DETAILS.

All courses are run at our Central London training centre and are also available on client-site.

**Call 071 922 8819 for course outlines and bookings.**

# RICHFORDS

SOUTH BANK TECHNOPARK • 90 LONDON ROAD LONDON • SE1 6LN • FAX 071 922 8839

# Letters

*We welcome short letters on any subject that is relevant to software development. Please write to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. Unless your letter is marked 'Not for Publication', it will be considered for inclusion in this section.*

## A case for JPEG

Sir,

Nice article on Fractal Compression, but there are some points that I felt were absent.

- Fractal Compression is very much slower than JPEG, as is decompression. Special hardware is needed to achieve useful performance - hardware which has a restrictive limit on the overall image size it can handle.
- JPEG technology is public domain, and available free in source form for both compression and decompression. On the other hand, FIF is proprietary, with hefty licence fees.
- While JPEG fidelity certainly degrades quite sharply at the high end of the compression range, a slightly less high compression gives you adequately small files (say 20 KB, instead of 10 KB), with acceptable quality.

Although high compression ratios tend to degrade JPEG picture quality, this compromise is quite acceptable, given my first two points.

*Don Milne*
*Aberdeen*

## Against Windows

Sir,

Am I paranoid, or is the world getting far too carried away with the Windows phenomenon? Is Windows really that good, or is the whole thing a Microsoft ploy to dominate the corporate desktop with slow, buggy software and copious amounts of hype?

You've heard the standard anti-Windows argument before, about the way that a 640 KB machine with a 20 MB hard disk is now no longer adequate for ordinary use, and that PCs now need 8 MB of RAM and a 100 MB hard disk to run Windows at any reasonable speed. But my experience is that this state of affairs is seriously irritating end users, as well as developers.

Take, as an example, a department that uses PCs to do word processing and a little database work. A couple of years ago, these people would have been using WordPerfect V5.1 and FoxPro, under MS-DOS, and would have got blindingly fast performance on 640 KB 286-based machines. But the hype generated by Microsoft, and carried on in the pages of all the mainstream computer magazines, have persuaded this department that its machines are hopelessly out of date and that it should upgrade to Word for Windows and FoxPro for Windows. This means upgrading the hardware too, at a hefty cost.

And what do our users get for all this extra effort and money? A system with a pretty user interface but which runs much slower, and is more difficult to use, than what they had before. Still, I'm sure that this is more than offset by the satisfaction that they are keeping up with their competitors and using modern technology.

Go on then, call me a cynical Luddite. I can bear it.

*Peter Dawes*
*Stockport*

## Nö, Nö, Nö!

Sir,

Now that Will Watts has vacated the Editor's chair, what will happen to the diareses (as in coördinate) of which Will was so fond? No one else that I have come across uses them, and they are such a typographical oddity that encountering one in the middle of a sentence invariably derails me and I have unnecessary difficulty reading articles in *.EXE* as a coherent whole. I just can't see that they are necessary. *Cooperation, coordinates* and so on are perfectly intelligible and non-distracting without diaresis or hyphen. On rare occasions a discrete hyphen may be necessary but the diaresis just jumps off the page and smacks me round the face saying 'hey, you don't want to read this article - look at me instead!'

*Chris Jolly*
*Text Systems Ltd*

## Coop or Coöp

Sir,

In defence of the diaresis, I draw on the remarks made by Sir Ernest Gowers on the subject in *The Complete Plain Words*: 'I have no intention of taking hyphens seriously. Those who wish to do so I leave to Fowler's eleven columns.... For instance, the general practice of hyphenating "co" when it is attached as a prefix to a word beginning with a vowel has always seemed to me absurd, especially as it leads to such possibilities of misunderstanding as unco-ordinated most present to a Scotsman. If it is objected that ambiguity may result, and readers may be puzzled whether coop is something to put a hen in or a profit sharing organisation, this should be removed by a diaresis (coöp) not a hyphen (co-op). That is what a diaresis is for.'

*rawkiw@cix.compulink.co.uk*

## Right to segment

Sir,

I was amazed, astounded, and generally gobsmacked by the two letters 'Flattening May' on page 10 of the July '93 issue. I'm being attacked for a single throw-away line in my piece about NT. I acknowledged that some people like flat memory, and some people don't. I said I don't like it. That's because I don't.

Contrary to what Mr Henney says, we live in the age of multi-tasking, multiple stacks, multiple heaps and overlays; it's called Windows. Why shouldn't I use such things in my programs? Microsoft is abandoning segments just as the hardware was reaching a point where segments were genuinely useful. That deprives both him and me of options.

Messrs Henney and Collins; if you like flat memory, please feel free to use Win32s or a Macintosh (or even, if you must, NT). But, where do you get off telling me I have no right to like segments?

*Jules May*

### Letter of the Month

***The writer of the best letter of the month, as judged by the Editor, will receive a .EXE disk of his choice. The best letter is the one printed first. Please note that letters submitted to this page may be edited.***

# So what about Aspen?

*Just when you thought that CA would never get around to updating Clipper, along comes CA-Visual Objects. But what exactly is it? Robert Schifreen tries to unravel the mystery.*

When Computer Associates bought Nantucket, no one knew what it would all mean for the future of Clipper. Would Clipper V6.0 be developed? And what fate would befall Aspen, the OOP version of Clipper that Nantucket had demonstrated at a developer conference shortly before CA got its hands on the company?

As it turned out, things went rather quiet for a year or two. We all assumed that Clipper's development would stop at V5.01, and some of us were beginning to doubt CA's promises that new and good things would soon emerge from CA's Clipper development team.

But just as things were starting to get far too quiet, up pops Visual Objects, a database development environment that can best be summed up as Clipper with Objects. The first version will run under Windows 3.1, with NT and OS/2 versions to follow. Is this Aspen? If not, what is? In the week when Visual Objects went into beta, we spoke to CA to get the full story.

## Typical Developers

The company is pushing the product in two major directions. While Visual Objects is 'a natural progression for xBASE users who want to develop Windows applications that benefit from object-oriented programming', it is also being heavily targeted at 'developers of the next generation of applications: mission-critical, financial and transaction processing applications that are the operational backbone of an enterprise.' If you thought that the Clipper group at CA would share the same business goals as Nantucket, think again.

CA seems to be jumping firmly on the objects bandwagon and is heavily promoting the product's ability to product object-oriented code using the built-in visual tools in VO. Yet research by Microsoft in the US towards the end of last year found that only 15% of VC++ users have seriously committed to using C++ rather than plain C. Has CA gone too far away from procedural code? The company doesn't think so. 'The visual tools in CA-Visual Objects generate object-oriented code. This makes future modifications easier and promotes consistency in user interface design,' it predicts. 'The graphical tools create MDI code automatically; other products do not. Each of the tools interacts with the CA-Visual Objects repository so the relationship between windows, menus, reports, action code and other resources is maintained constantly by the system.'

What exactly is the repository? Traditionally, a make file was needed to control application development. Because the active repository in CA-Visual Objects keeps track of all the relationships between application components, it is said to make building applications much simpler. The repository also provides incremental compilation with entity-level granularity, which means that an individual function may be recompiled instead of a whole file.

## Too New?

If you're starting to worry that VO contains too many new features, and will require you to change the way that you program, then don't. Visual Objects supports both procedural and object-oriented programming concepts. It is the first major business application development language with this capability, says CA, conveniently forgetting Visual C++ (although anyone who hasn't seen the class libraries and templates could argue that this is not an application development language). Mixing procedural and object-oriented programming allows developers to port legacy applications, thus helping CA sell more into corporate sites.

## Native Compiler

Visual Objects is, says CA, a true compiler, producing native 80x86 code rather than tokenised, interpreted routines. VO uses both native- and P-code, so developers can mix rigorous and loose programming practices, even within the same function. This means developers can continue to work as they have, while gradually tightening up their code.

## Clipper for Windows

Although CA-Visual Objects will take CA-Clipper programs and run them under Windows, says CA, that is only one facet of this 'robust application development environment.' So what happened to Aspen? Is VO actually based on Aspen? CA says that indeed it is. Or at least, it's the result of the Aspen project, which provided some of the key technologies. CA has added other major components, such as visual development tools, class libraries and SQL access.

## Competitors

What products are going to compete with VO? Presumably FoxPro and dBASE for Windows? Yes, concedes CA, but 'only to a limited degree'. FoxPro is not object-oriented, and neither product generates native machine code. This, says CA, coupled with the extensive class libraries and client/server support, sets VO apart. We'll have to wait and see whether this is indeed true - FoxPro is blindingly fast, native code or no native code.

EXE

# Geoffrey!

## Has someone deleted your copy of Underpants?

***Well, no, I cleared it off my hard disk, actually. Didn't seem any point in keeping it, now that most of my work is in Windows. I use ED for Windows all the time these days.***

Sounds like a rash decision, my son. I bet you'll miss your multiple buffers, configurable keyboard, C-like macro whatsit, undo/redo, multiple compiler support, templates, smart indenting.

***This is the 1990s, Brian.*** *Any* ***half-decent programmer's editor does all that stuff. I use ED for the extras: like colour syntax highlighting...***

You what?

***You Know.***

```
char * title = "Warning!",
       buf[10];
// pulls in dialog values
GetSearchStr(buf);
if (SQLSearch(db,buf))
{
  int MessageBox(this, "Sea
```

Very pretty. I suppose it does any language, as long as it's C++?

***Au Contraire. It'll do any language, as long as it's not APL. And you can define your own keywords. Look: I've got the Windows API set up, plus Jim's COBOL library.***

Talking of `SQLSearch(db,buf)`, could you load the source? I'm not sure, but I think you'll find it in U:\PDS\JIM\UPDATE\LIBSORCE\...

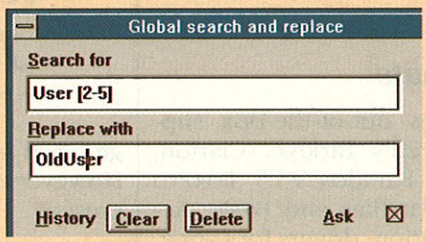
- Strewth! how did you manage that?

```
// C Interface for COBOL

// Implementation of SQLSearch
// Last modified Jim 2/9/93
int SQLSearch(DB* view, char *
{
  // Searches view database for
}
```

***Just right-button clicked on the function name. ED remembers where all your functions live.***

Hmm. Still, you can't beat good old grep...

***...unless you want to do a multi-file search*** *and* ***replace...***




Sounds more like a jukebox than an editor. How long did it take you to master this ED thing?

***That was the best part. ED is a smart editor:smart enough to work the way you do. Which mention of work reminds me: what was it you wanted, Brian?***

extensive colour syntax highlighting, smart language sensitive editing for all popular languages, code templates and completion, hypertext function lookup, bracket and object matching, compiler support with error tracking, emulates popular editors, named keyboard macros, regular expression search and replace, search files across all drives and directories (grep), file comparison, unlimited do and redo, context sensitive Windows SDK help, fast C extension language, LAN file-locking, printing with font selection, column and stream blocks, ruler, line drawing, multiple windows + files + buffers, comment alignment, Windows toolbar

Ed for Windows retails at £145. Ed for DOS retails at £99. ED for Windows with ED for DOS costs £199

***Don't Hack It - ED IT***

*For more information contact:*

# QBS

**QBS SOFTWARE LTD**

**10 Barley Mow Passage, Chiswick**
**London W4 4PH, England**
**Tel: +44-81-994-4842 Fax:+44-81-994-3441**
**BBS: +44-81-747-1979**

# A marriage across generations

*Clarion's 4GL incorporates compiler technology from JPI. Philip McNulty gives his first impressions.*

Clarion products, particularly Clarion Personal Developer for PCs are well-known and popular on the other side of The Pond, but have been fairly invisible here. After the recent merger of Jensen & Partners with Clarion Software in 1992, all that may be about to change. Clarion Database Developer V3.0 (CDD) is a database program/applications generator that combines traditional 4GL features such as extensive database support with native 80x86 code compiler technology from JPI. Although CDD is only available under DOS, Clarion has assured me that versions for OS/2 and Windows 3.x are well in-hand.

## File Formats

CDD provides out-of-the-box support for Novell's Btrieve, Clarion, dBASE III/IV, Paradox V3.5, FoxPro V2.0, ASCII flat files and Basic. According to Clarion, drivers for C-tree, Clipper, Netware SQL, Oracle and SQL Server will be available before the end of the year. Clarion has also developed its own database format (although Btrieve seems to be the default engine unless you select another option). It is multi-user, clearly groomed for Client-Server applications. However it lacks a SQL interface at present.

***Clarion Database Developer combines a traditional 4GL with compiler technology from JPI***

The back-end file drivers give considerable DBMS independence and do not restrict, in any way, multi-user access to the supported databases. All may be freely distributed without run-time licences. I have only tried the Btrieve and FoxPro database drivers shipped with CDD. Not all the features of Clarion are available on all networks, and some DBMSs are, of course, inherently single-user; dBASE III, for example. What Clarion provides is a single 4GL and procedural language interface to a wide variety and growing number of database file structures.

Clarion appears to be moving steadily away from its single-user xBASE origins to the new world of Client-Server architectures (Oracle and SQL Server being due by the end of the year).

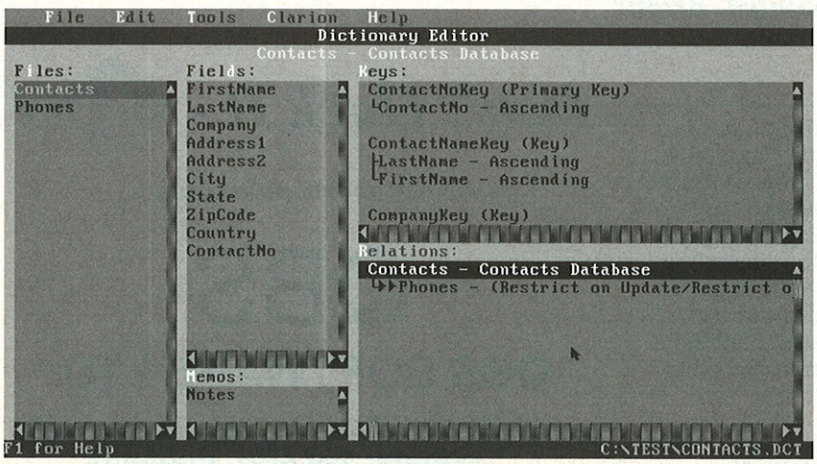



*Figure 1 - The Data Dictionary editor*

## Working in CDD

As you'd expect, CDD is project-based. There are two main components: a data dictionary editor which defines tables used in the application and the relationships between them, and an application generator with screen painters, report designers and menu builders for creating the program. The first thing you'll notice when CDD is run, is the IDE, with a menu bar for selecting various tools, invoking online Help and opening/building projects. The mouse cursor is of the smooth variety; this despite being run in DOS text-mode (for an example see Dave Mansell's *A Mouse's Tail* article, *.EXE* August '93). Furthermore, Clarion has defined intricate bitmapped radio buttons and check boxes which bear a striking resemblance to their GUI-based cousins.

As a labour-saving device alone, a good data dictionary is worth many

man-weeks of tedious error-prone work in database design and construction. The screen shot in Figure 1 shows Clarion's dictionary editor. It is divided into five areas (or worksheets in Clarion terminology) - File, Field, Key, Memo and Relation. To add a new table a developer simply supplies a new file name in the File worksheet then proceeds by specifying its fields (data type, format string etc).

One surprising feature of Clarion's dictionary editor is in the area of database consistency and integrity checking. You can choose between three automatic ways of ensuring how relational integrity is maintained in an application when primary keys are changed or deleted. You can prohibit the action, allow the action to cascade through all secondary files where the key appears as a foreign key, or you can automatically cause a null value to be set to the foreign key. Developers coming from a SQL background expect this, but coming from xBASE and Btrieve, it's a real bonus. Especially since it would have been necessary to hand-code integrity checking into such applications.

There is one particular productivity tool I for one would like to have seen complementing the dictionary editor. We have over 30 FoxPro DBFs on my present project. I don't relish the thought of typing these file/index definitions by hand into the Clarion data dictionary. So a utility program that will reverse engineer a DD from FoxPro DBF definitions would be nice. Checking the dictionary for relational normalisation etc would then be a small part of the task.

Once the data dictionary has been created, the next stage is the application's procedures, screens and menus. For this, it is necessary to bring up the application generator. Here you specify the name of a template file which CDD will use to generate code, and the data dictionary you have previously created. Any Clarion application must have a main procedure as a starting point for the application. This procedure will form the basis of the menu system or main screen in the application. A screen shot of the application generator is given in Figure 2.

## Clarion 4GL

The front-end to CDD is a powerful data dictionary-driven 4GL which generates Clarion source code. You can also add your own handwritten Clarion code, or you can write 3GL code in any of the Topspeed languages - Modula 2, C, Pascal or C++.

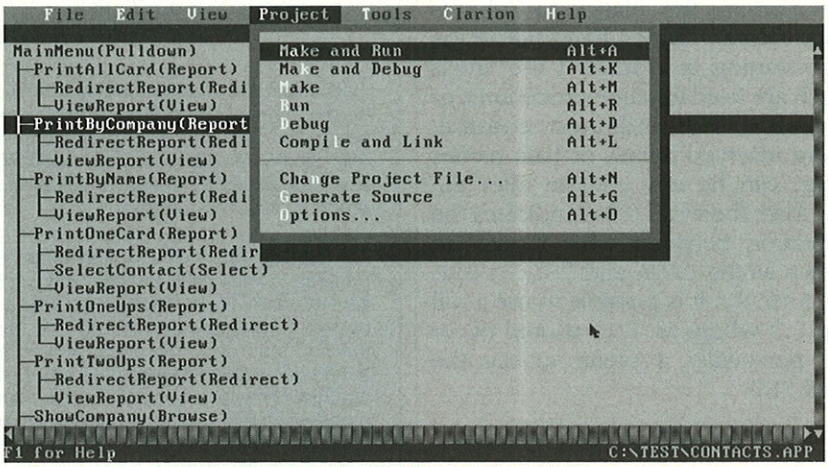



*Figure 2 - The Application Generator*

The heart of the 4GL is a metalanguage version of Clarion implemented as a set of procedural templates. The 10 templates provided with the package enable most basic database retrieve, update and browse functions to be created for an application without writing a line of code. Templates are also available from third party developers, so you can buy them in the same way that you can buy C++ Class libraries. Well, I'm no greater believer in re-inventing wheels, and buying source code metalanguage templates is as good a way of boosting programmer productivity as any. For a business-oriented language, metalanguage templates look a lot less intimidating to learn than mastering the OOP features of a language like C++.

The 4GL is a little more complete than the equivalent FoxPro Screen Generator. FoxPro provides only a System Menu generator and the Screen Generator, which, although very good, link with absolutely nothing else automatically. Clarion provides full menus, screen painter and report generator.

Clarion screens are also CUA/SAA compliant. This is an attractive feature for contractors like myself who frequently work in several different operating environments where we need to implement a single consistent end-user interface on projects. My latest fax from Clarion announced a new GUI kit to allow CDD users to mimic X-Windows.

## The Clarion Language

At first, I wasn't too wildly excited by the thought of having to learn a new programming language, but I found Clarion particularly pleasant. The statements, which are wordy, are mainly related to screen definitions. Like FoxPro's Screen Generator, the trick is to let the 4GL handle the tedious bits and use your skill in writing the start and end procedures linked to field or screen entry/exit processing. Following this approach, I was able to produce reasonably good Clarion code after only seven days' practice.

Superficially a Clarion program looks like a hybrid between COBOL and Modula-2. Impossible to imagine? In fact, the Clarion design team have come up with a pretty good fusion of both languages. Figure 3 lists a few skeletal snippets just to give you a flavour.

Notice that MYPROG2 was announced as being a separate MODULE in the MAP prelude of a program. Each module has a structure such as:

```
    ! a member of the
      ! main module
      MEMBER('XYZPROG')

Proc2  PROCEDURE
         CODE
         ! end of procedure
         END
         ! end of XYZPROG
         END
```

The map part of a program describes all the modules related to this program. After the end of the MAP section, the user can declare data variables until the word CODE is reached. Everything then to the last END statement is program code.

Now perhaps the relation to Cobol is a little clearer. The left-hand side of the page starting in column 1 are labels which are used to name procedures or variables. The indented column, where reserved names or data names begin, can be any column after the first. Also, the word CODE indicates the separation between what would be known as the DATA and PROCEDURE DIVISIONS. It is possible to use a full stop (.) where an END should occur but, personally, I prefer to see the word END.

Another Cobol feature inherited is the GROUP (ie unions in C). In Cobol, aliasing a memory area to another data structure is known as REDEFINITION expressed in the DATA DIVISION by a clause such as B REDEFINES A. Union declarations in C or record variants in Modula-2 and Pascal may be a bit more elegant from a coding viewpoint, but the way it is implemented in Clarion is geared towards report formatting and aliasing fixed-length strings: a feature which is best handled by the 4GL.

I have noticed the lack of one feature missing in the language: some form of generically independent query/filter facility to extract record subsets for local manipulation/reporting etc. The database filter facility allows one to set simple range checks, but I tried entering wildcards which it definitely didn't like. I have implemented an Informix style of filter using wildcards and range queries under FoxPro. It is a lot simpler than an RQBE method of entering queries; it is totally generic and the users can work through their standard input screens.

At present Clarion's proprietary language may seem a disadvantage. But I am rapidly coming to the conclusion that it could prove a winning card long-term. Clarion is simple to learn, and the difficult bits are so well handled by the 4GL that hand-coding them will appeal only to total masochists.

```
    PROGRAM
    MAP
        Proc1

        MODULE('XYZPROG')
      Proc2
        END
    END
    CODE
       ! here is the main body of code
PROC1   PROCEDURE
    CODE
        ! more code
    END  !end of procedure
    END
```

*Figure 3 - Hybrid Cobol & Modula-2*

## A Sycero Developer speaks...

Two years after I founded Newenden Associates in 1986, we made the move to Sycero dB as our primary Clipper development environment. We wanted a tool to speed up the generation of 'standard' screen, report and file handling code, and to assist with structuring and documenting projects. We decided on Sycero because it handled all these functions, was data dictionary-based, and was flexible enough to handle hand structured and coded programs where necessary.

The applications we've written under Sycero range from simple management systems running on small peer-to-peer LANs up to multi-site point of sale ticketing systems used by several large ferry operators around the UK. The ticketing systems control millions of pounds of revenue per annum and need to be reliable; using Sycero, we were able to develop them within budgets and timescales that would not have been achievable by hand-coding alone. Detractors from application generators often accuse them of inflexibility, but Sycero has proved capable of handling our requirements without causing any significant problems of this sort. Incorporating other Clipper code (written in-house or from third party libraries) is generally just as easy from within Sycero as in native Clipper.

Particularly important to a small software house is the consistency of coding which application generators can be used to encourage. Contract programmers are used for large projects, and Sycero has enabled Newenden to impose sufficient structure on contractors' coding to enable in-house staff to maintain it easily in the future.

What have been the problems? With Sycero dB, very few. Programs are slightly larger than they could be if optimised by hand, and there have been a very few occasions under older versions of Sycero where we've manually needed to tweak the generated Clipper code. A major gripe has been the long wait for V3.0 with full Clipper V5.0 support and features such as user-definable program templates, available in Sycero C for a long while. We have found Sycero's technical support to be very good. Minor updates and bug fixes have been provided promptly when required.

At Newenden, we have also used the C-generating Sycero C which, while excellent in concept, has not been problem-free in use. The very nature of the product makes it much slower at generating code than Sycero dB, and we've encountered several time consuming problems (eg with C memory allocation and deallocation - which had never been an issue under Clipper). Paradoxically, incorporating functions from external C libraries for serial I/O etc has proved very straightforward. The finished applications work well and are robust, but the main benefits of Sycero C will only be felt by those developing for UNIX and Xenix, as well as DOS.

*Chris Moore*
*Newenden Associates*
*Email: cmoore@cix.compulink.co.uk*

## Visual Debugger

VID is the source-level debugger which JPI used to ship with its compiler family of products (see Figure 4). You can set breakpoints, run trace, disassemble source code, set watch windows - all from popup menus with hotkeys. Well, I can't remember every use of F4 in life so popup menus are great *aides memoire*!

## Graphics Mode

Clarion has a whole list of in-built goodies which work under SVGA. The company has provided a powerful means of incorporating .PCX or .GIF images within your database and screen designs. Screens can be defined to be larger than their physical size: you can pan and scroll a screen by simply moving the mouse. Clarion also sent me a copy of its VGA kit for customising screen colours. Before that arrived my Clarion screens looked like a David Hockney painting.

## The Clarion API interface

By virtue of the fact that Clarion is written using the TopSpeed compiler technology, interfacing code written in TopSpeed C, C++, Pascal or Modula-2 (my favourite) is very easy. The Clarion API kit is primarily aimed at the C language market at present. It comprises header files and a book of advice on do's and don'ts. Compared with the FoxPro/Watcom API, which I found difficult to use, it looks straightforward. The interface for Modula-2 and Pascal users is even simpler than C because neither language relies on header files. You will need a TopSpeed C, C++, Pascal or Modula-2 compiler (ie Borland's or Microsoft's won't do) because all JPI compilers use the same optimising register passing conventions.

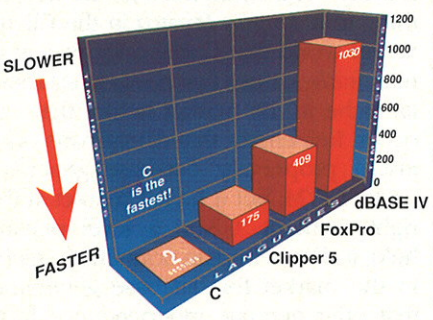

# Discover why FoxPro, Clipper, and dBASE were all written in C.

There is a good reason why your database language was developed in C. In fact, there are many good reasons.

C code is small. C code is fast. C code is portable. C code is flexible. C is the language of choice for today's professional developer. With the growing complexity of database applications, C is a realistic alternative. Now with CodeBase 5.0, you can have all the functionality, simplicity and power of traditional database languages together with the benefits of C/C++.

## C speed - fast code, true executables...

FoxPro, Clipper, and dBASE were written in C primarily for speed. But those compilers don't really compile, they combine imbedded language interpreters into your .EXE. Now that's slow. For dazzling performance you need the true executables of C. With CodeBase you get the real thing, C code. Consider the following statistics, from the publisher of Clipper:




"Sieve of Erastothenes"
Benchmark for Prime Number Generation
*Shows C to be incredibly faster !*

## C size - small executables, no added overhead...

FoxPro, Clipper and dBASE would like you to believe you need their entire development system to build database applications. But remember, those products are all written in C. So why do you need to lug all their extra code around? You don't. CodeBase is a complete DBMS, in C. No fat executables stuffed with unused code. No runtime modules. No royalties. Just quality C code. CodeBase is just what you need.

## C portability - ANSI C/C++ on every hardware platform...

No other language exists on more platforms than C/C++. Why rewrite your entire application for DOS, Windows, Windows NT, OS/2 or UNIX? With CodeBase the complete C source code is included, so you can port to any platform with an ANSI C or C++ compiler. Now and in the future.

## dBASE Compatible data, index and memo files...

You want the industry standard. You need compatibility. Sure, dBASE is the standard, but every dBASE compatible DBMS product uses its own unique index and memo file formats. Only CodeBase has them all: FoxPro (.cdx), Clipper (.ntx), dBASE IV (.mdx) and dBASE III (.ndx). Now it's your choice, we're compatible with you.

## Announcing CodeBase 5.0

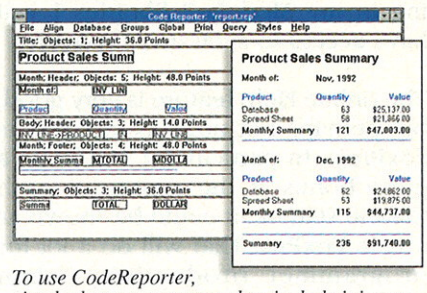
*The power of a complete DBMS, the benefits of C*

## NEW - Multi-user sharing with FoxPro, Clipper and dBASE...

Now your multi-user C/C++ programs can share data, index and memo files at the same time as concurrently running FoxPro, Clipper and dBASE programs. No incompatibilities. No waiting.

## NEW - Queries & Relations 1000 times faster...

CodeBase 5.0 now lets you query related data files with any logical dBASE expression. Our new Bit Optimization Technology (similar to FoxPro's Rushmore technology) uses index files to return a query on a 1/2 million record data file in just a second. Automatically take advantage of this query performance by using our new CodeReporter:




*To use CodeReporter, simply draw your report, then include it in any program you write. Call (071) 833-1022 now for your FREE working model of CodeReporter.*

## New - Design complex reports in just minutes...

Our new CodeReporter takes the painstaking work out of reports. Now simply design and draw reports interactively under Windows 3.1, then print or display them from any DOS, Windows or UNIX application.

## SPECIAL - FREE CodeReporter

Order CodeBase 5 before 31 May, 1993 and receive CodeReporter for free!

Distributed & supported by

**SEQUITER SOFTWARE UK LTD**
106-108 POWIS ST
LONDON SE18 6LU
TEL: 081 317 4321


**CodeBase 5.0**
*The C/C++ Library for DataBase Management*

**Call Now**
**081 317 4321**

## System Science

3-5 Cynthia Street, London, England, N1 9JF
**Tele:** +44 71 833 1022 **Fax:** +44 71 837 6411

Since JPI had a number of cross-company agreements with several companies marketing C and Modula-2 libraries, the API kit provides CDD users with a vast range of third-party tools and application add-ons. For example, we are using the Greenleaf Comms libraries on my present project.

## Docs & Help

Clarion comes with six manuals: *Tutorial, User's Guide, Programmer's Guide, Language Reference, Template Language Reference Manual* and the inevitable *Late Breaking News*. The manuals are clearly written and very informative but would benefit from some slight restructuring for the benefit of beginners. There must have been a host of callable library routines, but I wasn't able to spot where in the text you could call well-known functions like `Match(s1, s2)` (used to process a widcard match against a data field in a browse statement). At least by linking in my Modula-2 libraries I will have access to such useful goodies.

The online Help feature is very good, but perhaps not quite as inspired as FoxPro's. In particular it would be a great bonus if the syntax of each Clarion statement could be shown as it is in FoxPro. That will do a lot for programmer productivity when learning Clarion. For instance, although you can rely on the 4GL to do most of the work for you, on statements such as the `SCREEN` statement, hand-coding is not trivial.

Not only is the 4GL front-end comprehensive, and capable of delivering complete applications, it is also accompanied by a pretty comprehensive tutorial. I would recommend you allow at least two full days to work your way through the tutorial completely, since it goes through most of the 4GL features very thoroughly. You can stop at any stage you want and continue later. I also think that software companies like Clarion should take a leaf out Corel Draw's book and produce a video to accompany product tutorials (although Corel's was badly done).

***Remember: dBASE was once a proprietary language under CP/M, but it hit the market at the right moment***

## System Configuration

Clarion requires at least an 80286 CPU, 2Mb RAM, DOS V3.3 (or above). I would suggest that it needs at least a 60Mb hard disk if the development machine is also going to run the applications. Installation is from five 3.5" 1.44 MB diskettes. The full system takes up 14 MB of disk space. However it is possible to specify which chunks of CDD to install.

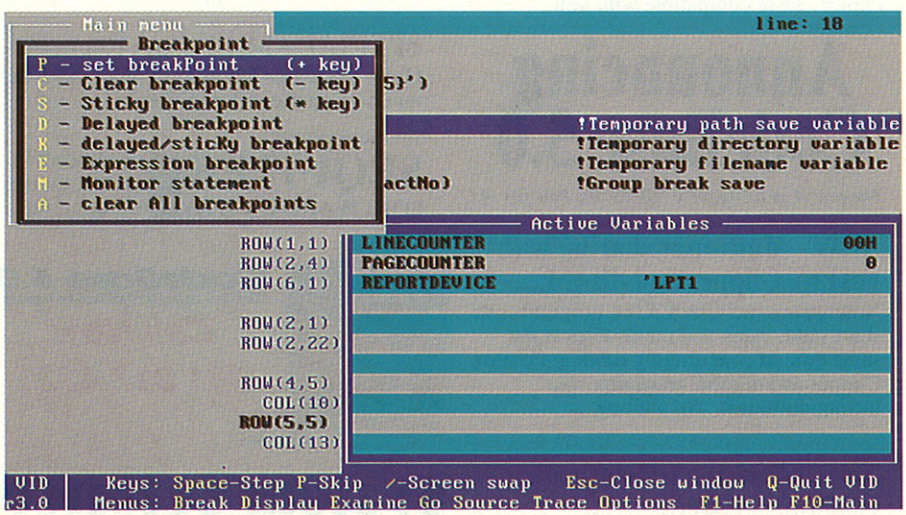



*Figure 4 - Debugging with VID*

## Conclusions

Clarion is a very powerful 4GL database application generator and language which is is fairly uniquely positioned in the market by being able to support multiple databases. It permits more than one type of database to be opened concurrently in an application. First impressions are that Clarion is fast and efficient. I'm using a 25 MHz, 4MB RAM 386 DX PC clone of somewhat uncertain pedigree, so my word is hardly authoritive. But the Sieve of Erasthostenes program provided with Clarion ran through produced over 300 prime numbers in 6.2 seconds, faster even than Borland C.

We need to get manufacturer-independent benchmarks on Clarion's overall performance in typical network environments to verify my initial impressions. Database applications performance in LAN-dominated environments is like an iceberg - 90% of the bulk is invisible - lost in the throttling affects of narrow LAN bandwidths and the inevitable I/O bottlenecks of remote servers. But I'm quite prepared to believe that the compiler efficiency is of a high order.

Clarion should appeal to independent developers writing applications software for their customers, and corporate users downsizing their existing applications currently running on mainframe/mini-computers. It should also be an excellent transition tool in managing the process of change to Client-Server architectures.

If Clarion gains the market recognition it deserves, it could become the nearest thing to a *Lingua Franca* in the Client-Server world that Cobol once was in in mainframes. But it first needs to establish its niche in the market before that can occur. Remember that dBASE was once just a proprietary database generator under CP/M; but it hit the market at the right moment. Given a level playing field, it should find an honourable niche in the market for database generators that offer genuine independence from totally proprietary engines.

EXE

---

*Philip McNulty has been working in computing for over 20 years. He specialises in real time, process control, LANs, and PC networking.*

Some people say that searching for OS/2 software is like looking for a needle in a haystack.

No more.

The OS/2 Solution Centre has changed all that. It is the place to go for virtually everything that runs on OS/2. With many OS/2 products, a library of OS/2 books, support, training and technical services, The OS/2 Solution Centre is the solution to all your OS/2 needs

### OS/2 software all sewn up

Programming tools, utilities, applications and every IBM product written - and if we haven't got it and it exists, we'll shift heaven and earth to get it for you.

Clip the coupon or ring (0285) 641175 today for our free catalogue.

### Specialist Training for OS/2

The OS/2 Solution Centre, as OS/2 specialists, gives you access to some of the finest OS/2 training courses available - courses that bring enormous benefits to attendees and your company.

### Technical services solving your problems

Although OS/2 virtually eliminates the most basic of computing problems, it's nice to know The OS/2 Solution Centre can provide you with access to specialists who can help when the more complex situations arise.

TO MAKE OS\2

SOFTWARE

EASY TO FIND

WE TOOK AWAY

THE HAYSTACK

### The support to keep you working

Access to the right level of support when you need it is key to the functioning of any organisation. The OS/2 Solution Centre has the means to help you resolve your problems.

One call and you'll be put in contact with a technical support specialist who will be able to help solve your problem - or put you in contact with someone who can.

For more information clip the coupon or telephone the OS/2 Solution Centre Hotline on 0285 641175.

## (A simple way of finding OS/2-based products)

NAME _____

POSITION _____

COMPANY _____

BUSINESS TYPE _____

HARDWARE CONFIGURATION _____

ADDRESS _____

_____

_____

TEL _____

FAX _____

EXE 10/93

THE OS/2 Solution CENTRE

Barton House, Barton Lane, Cirencester, Glos. GL72EE. England
Tel: (0285) 641175. Fax: (0285) 640181.
Int tel: 44 285 641175. Int. Fax: 44 285 640181

➤ CIRCLE NO. 069

# From GLs to GTIs

*Patrick McParland discusses the evolution of the 4GL, and why it is time developers took it seriously.*

Over the past 20 years, many companies have introduced computer systems to replace tasks previously carried out manually such as payroll packages, stock control systems and simple filing systems etc. Many industries have benefited from this automation of their administrative and manufacturing tasks. Today, this technique is being applied to the computer industry itself: the software industry is trying to automate its main manufacturing task - the development of software - by the introduction of software tools. The vendor hype surrounding these software tools is slowly changing our perception of software development from programming language concerns (ie Generation Languages or GLs) to tool use issues (ie Generations of Tools and their Integration - GTI).

## History of GLs

The term GL (or Generation Language) only came into being retrospectively with the wide scale adoption of 3GLs like Cobol and Fortran. Fortran was the first 3GL. It started life in the mid-1950s. Since that time, virtually all software development has concentrated on the use of a programming language. However, the obsession the software industry has with programming languages has become unhealthy. Developing applications using a purely 3GL approach has the following problems:

- Developers constantly work at the fine-grain level of detail associated with a programming language, making them less likely to generalise the current problem under consideration to produce a piece of code which could be reused in other applications.
- Developers are encouraged to write every application from scratch, believing that, because the low-level detail of their application is unique, then no portion of an application is reusable.
- Large 3GL applications are extremely difficult to maintain because the maintainers are often presented with a large body of 3GL code without any design information to help them identify the small portion of the code which must be changed.
- To encourage some abstraction away from the narrow confines of programming to the consideration of the end user requirements, development teams are often split into programmers and analysts. Ensuring that the two groups work as a coherent team is a major source of problems in many projects.
- The concentration on low-level programming concerns, the lack of reuse and the need for programmers and analysts, leads to long development times.

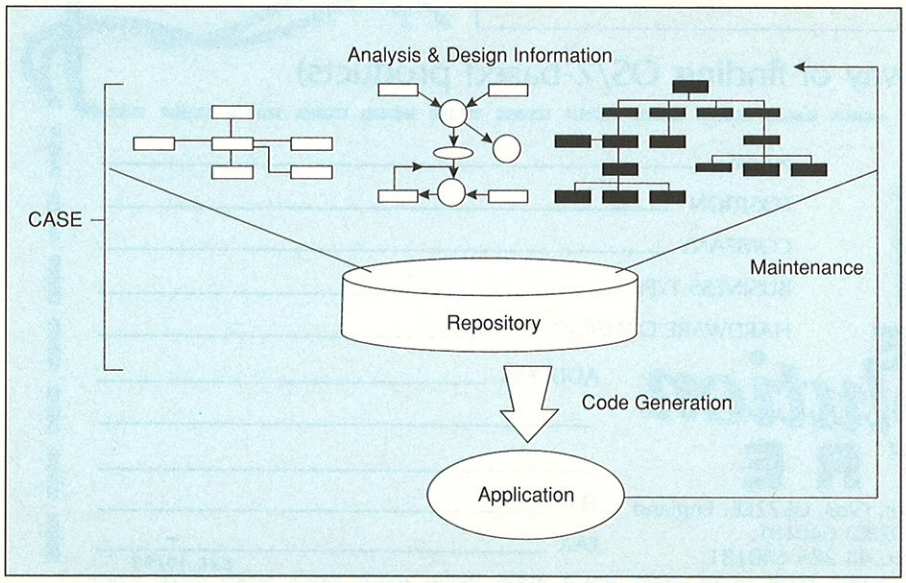



*Figure 1 - Relationship between a CASE tool and an Application*

However, these problems do not mean that developers should stop using 3GLs. For the development of intricate low-level systems there is no alternative. Since their conception in the 1950s developers have used 3GLs to build increasingly more sophisticated applications. Developers should not abandon this experience. However, the use of software tools can complement the use of 3GLs by increasing productivity, encouraging reuse and helping to control the software development process.

## Early Tool Use

Code generators have been available since the 1960s. They are able to infer part, or all of, the code implied by some form of requirements statement. So they would be able to generate a COBOL data division from a data specification. The initial examples were quite basic and received a bad reputation due to quite ludicrous oversell by the vendors - some things never change! One notorious example was called 'The Last One' which implied

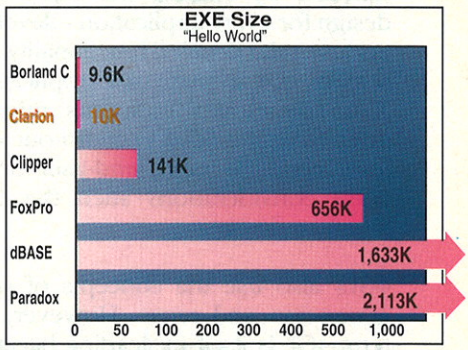
New Version 3!

# If You're Wondering About the Latest In Database Developer Technology...

# This Should Bring You Up To Speed!

***Faster, More Powerful, More Versatile • The Clarion Database Developer™ Version 3***

We know it's hard to believe, but at last, everything you've ever wanted in a database development platform is available now in the NEW Clarion Database Developer, Version 3.




## C Language Performance

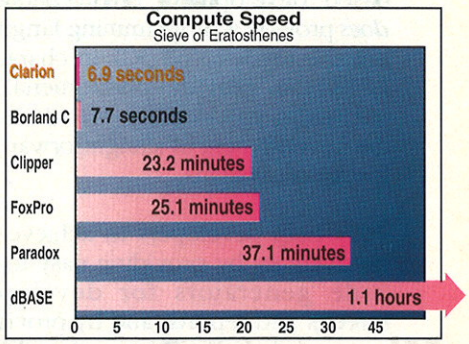
Our optimizing compiler and "smart" linker deliver run time performance that matches the best C compilers. "Hello World" takes only 10K. Benchmarks like the "Sieve of Eratosthenes" run 200 times faster than FoxPro and Clipper. The compiler produces standard .OBJs that can be linked with other languages and debugged with our own source level debugger. (See charts)

## Leading Edge Memory Management

Break the DOS 640K barrier with our automatic overlay system. Segment your application into DOS DLLs that can be distributed separately. Or run your application in protected mode with our optional royalty-free hardware assisted DOS extender-the ultimate memory manager.

## Awesome Video Effects

Create SAA/CUA compliant applications with pull-down menus and attractive graphic check boxes, radio buttons, and smooth mouse cursor. Place graphic images of any size on your text screens. Use any text mode: 80x25 to 132x44. Design virtual screens up to 16 times larger than normal. Open exploding or falling windows with see-through areas and transparent shadows.




## Replaceable Database Drivers

A universal data dictionary works with Clarion database drivers to produce applications for any database. Xbase, Paradox, Btrieve, c-tree, ASCII, and Clarion drivers are standard. SQL databases are available at an additional cost. Changing a single parameter ports your application to a different database.

## Interactive Visual Design

Assemble your application from our catalog of visual procedure templates then "paint" in your custom screen and report layouts. Embed your own source wherever it is needed. Then we generate commented pre-tested source code for you.

## Optimistic Concurrency

Clarion applications support optimistic concurrency to maximize your multi-user throughput and referential integrity to safeguard your data. These strategies work on any database concurrently with any application.

## No Royalties or Runtimes

Your applications compile into .EXE files. No royalty payments, no run-time modules, no copyright notices, and no LAN Packs or Bump disks.

| Product Comparison | dBASE | FoxPro | Paradox | Clipper | CLARION | Borland C |
|---|---|---|---|---|---|---|
| Visual Design | Partial | Partial | Partial | No | Complete | No |
| Data Dictionary | No | No | No | No | Yes | No |
| Database Support (Read/Write "in place") | Xbase | Xbase | Paradox | Xbase | Xbase, Paradox, Btrieve, SQL, etc... | None |
| Royalties or Run-times | Yes | Yes | Yes | No | No | No |
| Network Support | w/LAN Pack | w/LAN Ver. | w/LAN Pack | Unlimited | Unlimited | No |
| Memory Models | Overlay | Overlay Protected | Overlay | Static, Overlay | Static, Overlay DLL, Protected | Static, Overlay |


*Introducing the New Clarion Database Developer Version 3*

CLARION SOFTWARE

Yes, I Want To Be Brought Up To Speed!

*Please send me information about Clarion NOW.*

Name: _____

Company: _____

Address: _____

_____

_____ Post Code: _____

Tel: _____ Fax: _____

For a FREE Full Colour Brochure, please post or fax this coupon to:
**Clarion Software (Europe) Ltd.,** 3 The Mansards,
Tavistock Street, Bedford, MK40 2RX, England.
Fax: +44 (0)234 217094 or Call: +44 (0)234 267500

EXE 10/93

➤ CIRCLE NO. 070

that it was so comprehensive that it was the last computer program that would ever need to be written.

However, this technology has now matured with the availability of more comprehensive code generators (such as APS and Telon for Cobol applications). These application generators function by reusing the same core design for every application - developers insert the detailed functionality and screens for a particular application. Thus developer concerns migrate away from the need to churn-out low-level code, to more end-user orientated considerations such the user interface design.

4GLs also use the concepts of code generation and reuse. However, the term 4GL is itself misleading because, unlike a 3GL, a 4GL is not just a programming language but a tool-based development environment. It *does* provide a programming language but this tends to be 3GL in character. 4GLs can provide phenomenal increases in productivity over 3GLs for the development of straightforward information systems.

They help developers to achieve this productivity by providing easy to use code generators for developing screens and reports, and by providing a comprehensive library of verbs, or subroutines, for commonly used program functions. The result is that developers often only need to write code for the most complex processes in an application.

Developers can spend more time ensuring that the screens and the reports they produce meet the end-user's requirements. They can even use the 4GL to produce a prototype which the developer can discuss with the end-users. Thus developers spend less time coding and more time ensuring that the application satisfies the end-user's requirements. In some cases developers spend so little time writing detailed code that the need for separate analysts and programmers disappears.

However, 4GLs are not general purpose software development environments so they are not applicable to all software problems. In fact, some companies have had extremely bad experiences of using 4GLs on the wrong type of problems. 4GLs are excellent for the development of relatively simple information systems.

## The New Messiah

If you believe all you hear, then CASE (Computer Aided Software Engineering) tools are the latest, and the best, panacea for all your software development problems. Certainly they offer more support for the software developer than is available using 4GLs or code generators. A CASE tool supports a software development method that guides the developer through all stages of an application's development, from planning through to construction.

***One notorious code generator was called 'The Last One' which implied that it was the last computer program that would ever need to be written!***

Moreover, if the CASE tool provides a code generator to automate the construction phase, then the CASE tool can reduce significantly future maintenance problems with your applications. The CASE tool achieves this by maintaining a dynamic relationship between the application's code and its specification (see Figure 1). Developers make all updates to the application via the CASE tool's analysis and design tools and then regenerate the application's code using the code generator. Thus the application's specification is always up-to-date with respect to its code. Applications are easier to maintain when a specification is available.

However, perhaps a CASE tool's biggest aid to software development is the way it encourages developers and maintainers to concentrate their efforts on the analysis and design of an application. Instead of spewing out reams and reams of 3GL code, developers concentrate on understanding the end user's needs and then designing an application to satisfy those needs. Constructing the final application has a lower priority since developers can use the code generator to generate a large portion of the code automatically. As a result of this change in emphasis, some companies are dispensing with the need to divide development teams into analysts and programmers.

The use of CASE tools can also encourage reuse by allowing developers to work at the design-level where they can more easily identify common parts of different applications - leading to better code reuse.

## Other Tools

Tool use does not stop at the traditional boundaries of CASE technology - analysis, design and construction. They are available for project management, testing, version and configuration management, reverse engineering and many other areas. Future generations of software tools will support the integration of these tools into a single development environment.

There are two main approaches available for developers to achieve the level of integration they require: single vendor I-CASE (Integrated CASE) and C-CASE (Component CASE). Using the I-CASE approach the developers choose a software tool vendor offering an integrated set of tools (IE analysis and design tools, code generators and project management tools) and acquires all of these tools. In theory, since all of the tools are from the same vendor, integrating them, to allow free and easy exchanges of information, should not be an issue. The problem with this approach is that no single vendor can provide the best tool support in every field of software development. Therefore, the price of integration can be the use of second class tools in several areas.

With C-CASE, developers select a standard repository (or data dictionary) and acquire tools from a range of vendors who have tools that interface with the repository (see Figure 2). This is the approach advocated by standards bodies, such as ANSI with

# Before you hatch your next system design, make sure it'll grow into something you can handle.

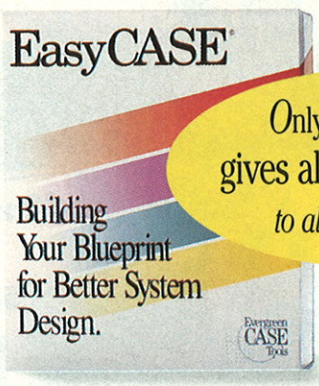
Born of hot deadlines and small budgets, poorly designed software applications can grow into monsters of complexity, eating up your IS department's time and money. Now, with new EasyCASE® 4.0, you can afford to give your *entire team* the powerful CASE tool it needs to build "blueprints" for developing high-quality, well-documented, maintainable systems. Whether it's a real-time or information system, EasyCASE 4.0 keeps you in control.

***Arm yourself with all these methodologies.***

EasyCASE 4.0 offers the *largest* selection of structured analysis, design and data modeling methodologies available in *one* tool: Yourdon/DeMarco, Gane & Sarson, Ward-Mellor/Hatley, SSADM, Yourdon/Constantine, Martin, Jackson, Bachman, IDEF1X, Shlaer-Mellor, Chen, and Merise.

***Ensure sound design, team-wide.***

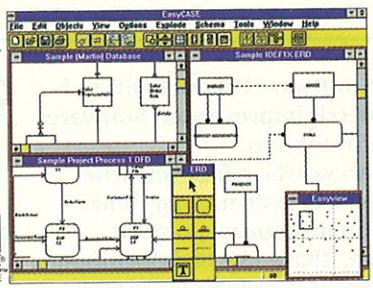
Our rule-based methodology monitor walks your entire team through development tasks in a consistent, uniform manner. It catches and prevents errors as you go. Database design is improved with automatic schema generation for xBASE and SQL, plus reverse engineering of xBASE.




Only EasyCASE gives all this *power* to all your people.

***Easy to master in Microsoft® Windows™ 3.1 or MS-DOS®***

Enjoy a consistent graphical user interface; object-oriented editing; easy chart linking; and on-line access to the integrated data dictionary. It's easy to learn. First-time users become productive quickly with a tutorial, extensive on-line help, methodology guide and more... all at a price you won't have to wrestle with.



*For more information, call:*

*071 833 1022*

***EasyCASE Professional 4.0***
***£645*** *for MS-DOS +VAT*
***£699*** *for Windows +VAT*

***EasyCASE System Designer 4.0***
*(Includes schema generated for xBASE and SQL)*
***£945*** *for MS-DOS +VAT*
***£1045*** *for Windows +VAT*

Evergreen CASE Tools

**System Science**
**3-5 Cynthia Street, London N1 9JF**

the IRDS repository, and hardware vendors such as IBM (AD/Cycle), Digital (Cohesion) and ICL. The C-CASE approach would allow developers to choose the best tools available and support their integration in a single development environment with the standard repository at its centre. The problem with this approach is that the sophisticated repositories necessary to support C-CASE are not widely available yet.

## Conclusion

Regardless of whether I-CASE or C-CASE becomes popular, a key concept in the 1990s will be the **capitalisation** of software developers. Developers will require the latest workstations with an integrated development environment using the latest software tools. Certainly, one of the easiest ways to increase productivity and quality is to encourage the use of software tools. Software tools enforce a standard notation and carry out many consistency checks which makes their use a valuable asset in the development of an application. However, introducing a software tool can be non-trivial as it may require major changes in the way a software development organisation operates.

Thus the vendor hype surrounding these new software tools will slowly change our perception of software development from programming concerns to design issues. Vendors are encouraging companies to buy more software tools and so capitalise their software developers. New programming languages will evolve and Cobol and C will live on but vendors will always have a new tool to market.

EXE

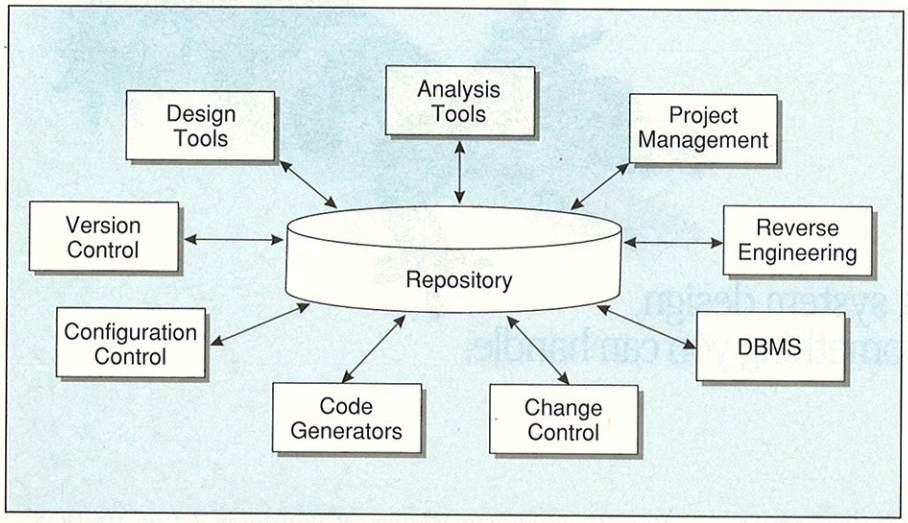
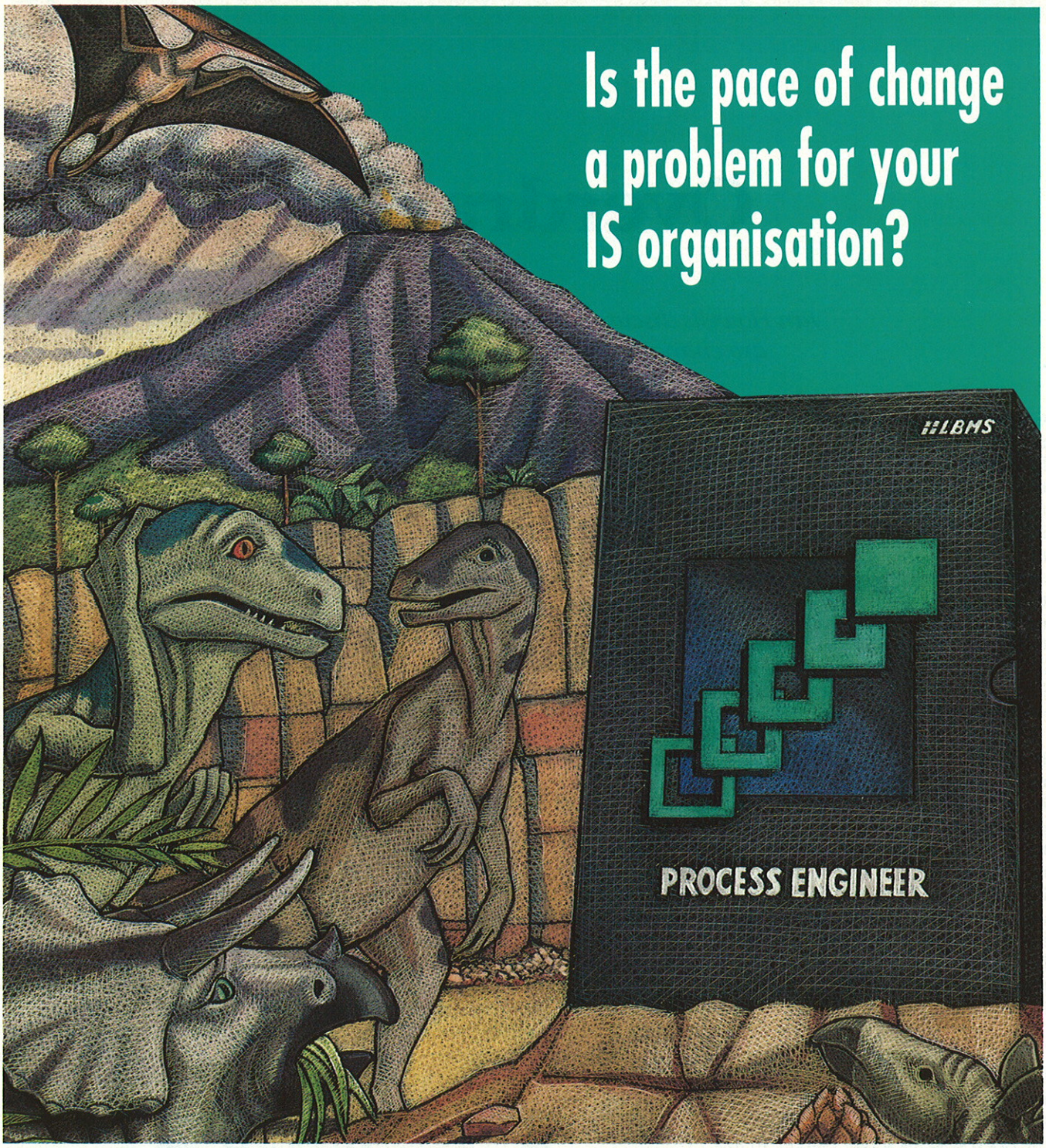



*Figure 2 - Integration of software tools with Component CASE*

*Patrick McParland is a lecturer with the Queen's University of Belfast and also works as a consultant for the software industry. He specialises in the introduction of CASE tools and recently completed a report on the use of CASE tools.*




## Paper-based standards inhibit change ...

**THEY** take forever to update and by the time you've finished they are probably out of date again. Today's business pressures and rapidly changing technology call for readily adaptable system development, project management and quality management processes. Process Engineer from LBMS is an automated process management environment that enables you to integrate and control all of these processes.

Your processes are defined in a configurable process library. Powerful software support makes updating the library simple and speeds up project estimating and planning. Process Engineer comes with a full set of state of the art application development and project management processes for you to adapt - fully documented in an intuitively simple hypertext guide.

Process Engineer is the only comprehensive process management environment that allows you to define, deploy, measure and improve your IS organisation's processes, whether they are proprietary or developed in-house.

LBMS is holding a series of seminars to show you how Process Engineer can revolutionise your ability to keep pace with change. To reserve your place please call **Marie Murphy** on **071-636-4213** .

**Seminar dates:**

**London** 29 Sept
**Manchester** 6 Oct
**London** 13 Oct
**Birmingham** 20 Oct
**Leeds** 27 Oct
**London** 3 Nov

LBMS

# Overdrawn!

*Ian Howells discusses how open standards and GUIs are changing the face of traditional OLTPs.*

Online Transaction Processing: the technology that forces your bank account to be debited when a cheque is cleared; that sits behind the 'hole in the wall' Automatic Teller Machine (ATM) insisting you remember your PIN number; that causes the terminals of the stocks/shares/commodities/currency traders to get updated with the most up-to-the-minute information. You'll find OLTP on systems which comprise a large number of users performing a large number of queries on databases with a large number of records.

Batch-processing is unsuitable for many of the applications of OLTP. For instance, if a bank used a batch program run at midnight to update all transactions of the day, your balance may say that you're overdrawn when, in fact, you're in credit. Hence, more and more processing cycles are moving from batch to OLTP. So many businesses need to have their applications based on OLTP technology instead of their current systems.

The move towards Open Systems OLTP has been driven by several large industry trends including decentralisation, rightsizing and the maturing of the open systems server market. In this article I will discuss the need for standards-driven architecture, and the great marketing opportunity which has arisen as a result of wide spread adoption of modern GUIs such as Windows. The reasons to move to GUI-TP and the principles behind it will also be examined.

## The Market

Open Systems OLTP is growing rapidly. It is currently estimated to be an industry of a size in the range of $2.4 to $4.4 billion. (InfoCorp estimates it will grow to a $15 billion market by 1996.) Traditional TP environments were developed for mainframes: consequently, they are 3GL character-based solutions. They were very successful at high performance transaction processing, but were limited in what facilities could be offered to the user. The advent of the relational DBMS brought with it, not only flexible, data storage and retrieval mechanisms, but also far more productive toolsets. What are now desperately required are powerful, productive tools, now commonplace in the RDBMS industry, that can be used against modern Open Systems TP monitors. In other words, the best of both worlds - cost-effective performance and flexible, along with state-of-the-art toolsets.

***Practices correct for the architectures of over a decade ago are still in use today***

## What's it all about

An OLTP is built on large (several gigabytes), shared databases with thousands of users. Queries are usually short and simple ie 'debit my account **now**'. However, a large number of such queries are processed by the system. Generally, I/O to and from the OLTP is well-defined. For instance, with the ubiquitous cash dispenser (ATM), there are inputs to withdraw cash, find balance, order statement/cheque book etc. It is important that transactions can be 'priority ordered'. When a bank detects fraud, it needs to run the 'shut down system' transaction immediately, and not have to wait for the transaction queue to clear. From the types of application-area which use OLTP it should be obvious that these are 'mission critical' applications. Consequently a high level of reliability is crucial.

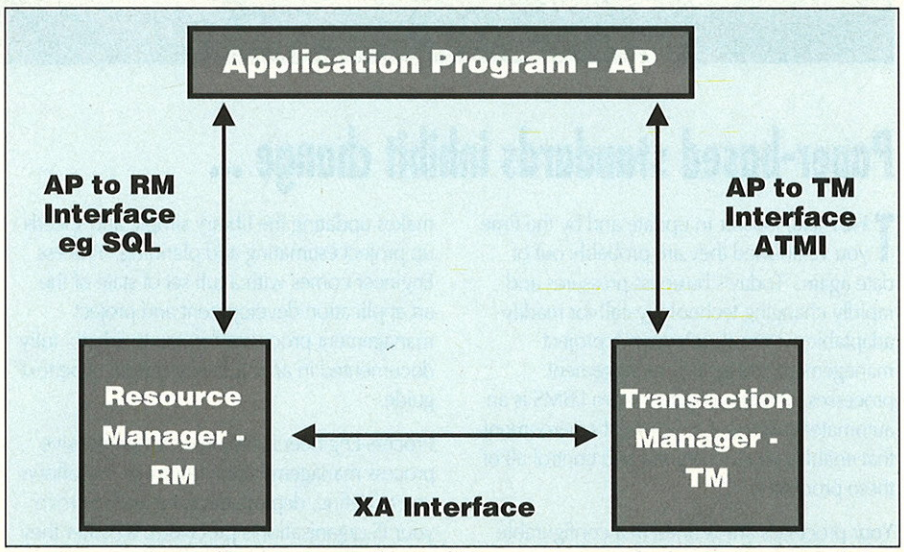
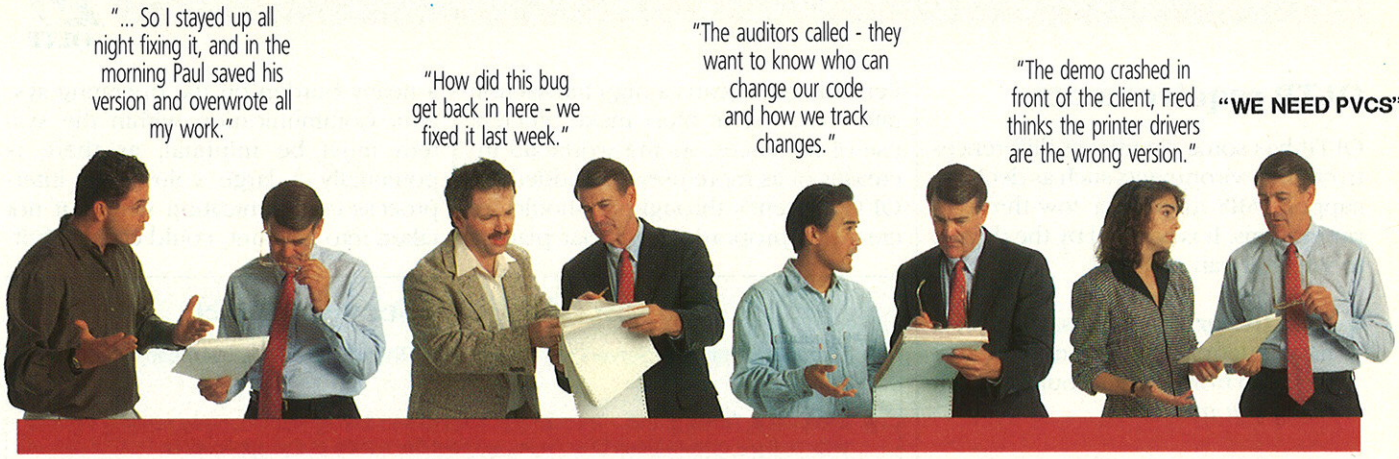



*Figure 1 - X/Open Distributed Transaction Processing model*




# Protect Your Code, Your Projects, and Your Reputation

Our customers know PVCS saves them money, time, and anguish. The painful scenarios PVCS avoids are familiar to all developers. Because *all developers do configuration management.* If you manage more than a few modules of code, have more than one person working on a project, or do LAN-based development, then you already do it. *The only question is how well.* And the answer affects the reliability of your projects, the productivity of your developers, and your reputation with your users.

The PVCS Series offers a complete solution to configuration management, that's important because development is getting more complex – not less. Implementing a partial solution today is simply creating tomorrow's problem.

**PVCS Version Manager** oversees changes to the elements that comprise a software system, tracking who made changes, when and why they made them, and permitting any previous revision of a module or any prior version of the entire software system to be reconstructed at any time. PVCS Version Manager also includes facilities to control parallel development and to reconcile any conflicting changes made to a module. PVCS enables a productive workgroup approach to development, because it is specifically architected for the LAN, and because the feature set and GUI interfaces specifically support distributed client server development. Modules are retrieved for work from wherever they are stored – developers don't have to remember where the archives are located. This simplifies and encourages code reuse, even across operating system boundaries.

**PVCS Configuration Builder** automates execution of software development tools, permitting systems to be built with absolute repeatability. This increases productivity, but more importantly, entirely eliminates a common source of errors. PVCS Configuration Builder can now embed "footprints" of vital historic information into compiled objects and can act conditionally on footprint information.

**PVCS Production Gateway** synchronizes the mainframe library to the LAN version archives, providing the control and auditability of the mainframe without bottlenecking the productivity of the LAN environment.

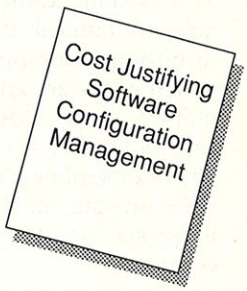
**PVCS Developer's Toolkit** delivers full PVCS functionality to your development environment or to your applications in the form of DLLs, import libraries and header files. The scope and quality of the PVCS DTK meets IBM and other programming standards for API implementation.

**PVCS Reporter** lets you quickly produce reports using an advanced graphical interface (in MS-DOS or Presentation Manager) to access the wealth of project information managed by PVCS. You can use predefined queries or build your own. PVCS Reporter queries can span multiple projects in distributed architectures to select, merge, sort, display and print exactly what you need.

PVCS is the undisputed leader in Configuration Management for LANs. It operates seamlessly across MS-DOS, OS/2, VMS, and a host of Unix environments.

*The PVCS Series is so accommodating that it snaps right into existing projects. You don't have to wait until you start a new project to implement it. More than a few of our customers have purchased PVCS to salvage projects that were dissolving into chaos... and they've been successful. Cure your biggest headaches before they happen.*

*Call now for a FREE copy of "Cost Justifying Software Configuration Management".*




Phone or fax for details of the PVCS Services. Tel 071 625 5255 Fax 071 624 9404

**Evaluation Copies Available**

**Readmar Systems**

L I M I T E D

239 Kilburn Park Road London NW6 5LG

New Windows / OS/2 GUI Version 5.1

➤ CIRCLE NO. 074

## OLTP requirements

OLTP has some significant differences to other environments such as decision support, MIS and other low-throughput systems. It is typified by the degree to which it can provide:

- **Data Integrity** - A transaction must have a definite goal. So money can't be taken from *your* account without being put into *mine*.
- **Control** - It should be possible to configure the capacity of transactions processed for different system loads.
- **Recoverability** - It is all very well taking backups, but we also need to restore from backups as quickly as possible: imagine what would happen if a banking system was 'down' for 24 hours.
- **Performance** - Throughput should increase as the number of users on the system increases.
- **Standardisation** - OLTPs are inherently complex. Better to build from non-proprietary components which work together.

Data integrity must be guaranteed by the service provider even in the case of system failures, including hardware failures. Such techniques as disk mirroring, dual ported peripherals and resilient software should be utilised to enable this.

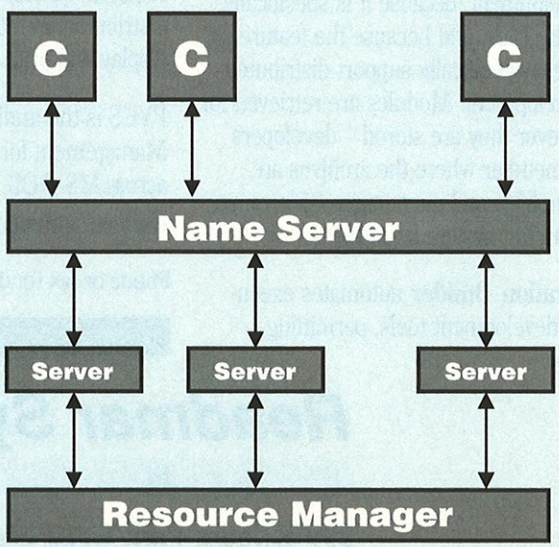
The system administrator should be able to control the system without disturbing running applications. Potential hardware and software must be utilised to provide resilience in the case of failure. Transaction integrity is vital. Remember: the more a company relies on data, the more it should invest in protecting it and maintaining its correctness.

Recovery/backups must occur continuous with the normal running of the system. Applications must not be halted to perform backups. Backups must cause a minimal performance degradation to the system and take place in a reasonable amount of time. Recovery should be rapid. Backups are only taken to facilitate recovery. Hence when recovery is needed the system should return to an on-line state as quickly as is possible.

Performance means a high transaction rate. The system must make efficient use of resources. As the workload increases or as more users are added, an OLTP system's throughput should increase in proportion, without placing a heavy burden on the operating system. Communication within the system must be minimal, as there is potentially a large volume of inter-process communication which, if not taken into account, could cause a bot-

### Enhanced Client/Server Model

- **High performance** - Servers run continuously. There is no startup overhead for a service request.
- **Increased Throughout** - Throughput is increased through continuously running servers and multiple occurrences of services to meet the needs of the application.
- **Location Transparency** - The name server carries location information. Client processes simply request a service by name.
- **Robustness** - Transaction control ensures database consistency. Servers can be restarted or moved to alternate locations as necessary.
- **Flexibility** - Dynamic control enables an administrator to tune the system to fit the traffic. New application functionality can be added without disturbing existing portions.
- **Scalability** - The number of servers or systems can be adjusted according to the needs of the application and the workload.
- **Efficiency** - Resource sharing reduces cost per user. Client processes connect to the name server through the Transaction Manager which provides location transparency (ie the location of the device is immaterial).
- **Interface Transparency** - The location of the processing of the client interface is transparent.
- **Transaction Capture Device Transparency** - The device that captures input is immaterial. Multiple different devices can concurrently be used.
- **Resilience** - Transaction capture device failure has no effect on the rest of the application.
- **Multiplexed Input** - Several Transaction Capture Devices can be combined into one input stream.
- **Smaller Client Processes** - Client processes can utilise a very small amount of memory. Server processes connect to the name server through the Transaction Manager. This provides location transparency (the location of the server process is defined in the configuration files and hence is transparent to the client processes).
- **Administration** - TP monitor facilities aid in system administration.
- **Load Balancing** - Dynamic load balancing can take place across Resource Managers.
- **Server Restriction** - Processing can be arranged by the arrangement of services within servers, by use of priorities or by sending jobs to a background queue.
- **Robustness** - Failure of a single application resource is well contained. Servers can be automatically restarted. Groups of servers can be reallocated to alternative machines in the event of a system becoming unavailable.



tleneck. The system must be reliable with a high availability. Thus, performance monitoring and management is required to tune the system and optimally use critical resources.

Having examined the requirements at the system, application and service level I will now look at standardisation in greater detail.

## Open Standards

Within the new Open Systems TP environment, standards have an important role. If we are to believe what we hear: 'they provide the framework for the future.'

X/Open has been working on defining an Open Distributed Transaction Processing model. The individual components illustrated in Figure 1 make up an X/Open conformant TP system, along with the operating system, database management system and other system facilities.

The Application Program (AP) component defines a transaction and uses resources within transaction boundaries. Each AP specifies a sequence of operations that involves resources such as terminals, (end-user interfaces) and database primitives such as `access` or `update`.

A number of Resource Managers (RM) can exist which manage some of the computer's shared resource. Other software entities can request access to the resources using an interface that the RM provides. Examples of RM's are: relational database management systems, Indexed Sequential Access Methods, print servers etc.

The Transaction Manager (TM) takes care of global transactions. This relates to transactions involving more than one RM. To enable this, it monitors the state of the global transaction and performs global failure recovery. Transaction managers can exist on multiple systems, communicating via the Distributed TP (DTP) Protocol. The X/Open Model defines two programming interfaces for the TP application programmer. SQL is used for accessing the database, and the Application Transaction Manager Interface (ATMI) provides an API for invoking the TM from application code.

***A transaction must have a definite goal. So money can't be taken from your account without being put into mine***

DTP is a protocol which allows two or more TMs to communicate and distribute information about transactions. XA is the interface between the TM and the RM (DBMS). It is used to send information about the transaction (the transaction id), and to perform the Two Phase Commit (2PC). It is the complete set of protocols that defines an open standards-based TP environment.

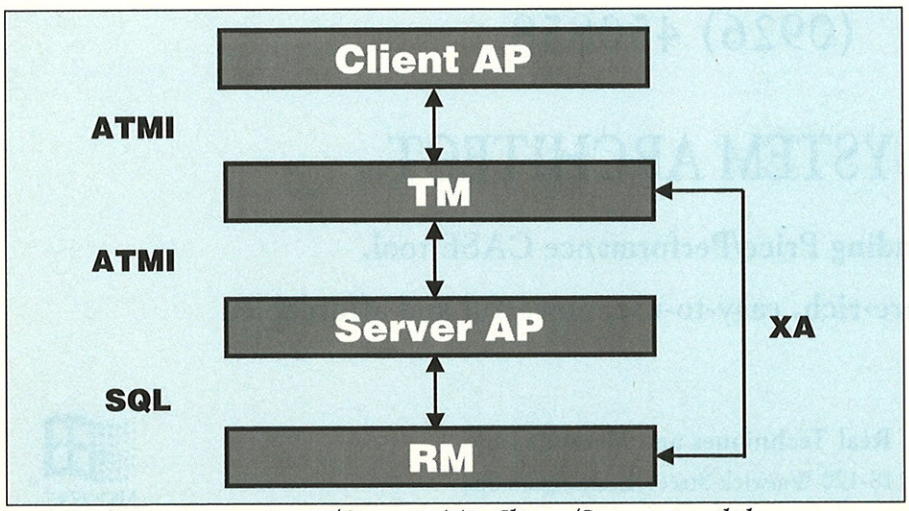



*Figure 2 - X/Open DTP Client/Server model*

## Client/Server

The advantages of client/server architectures, such as scalability, are well known. The enhanced Client/Server architecture improves on the 'basic' model by placing a 'Name Server' between client and server processes, and removing the requirement for each client to have a dedicated server. The advantages of this architecture are listed in the 'Enhanced Client/Server' box.

The relationship of the Client-Server Model to the X/Open Model is shown in Figure 2. As you can see, the application code is divided into two separate application processes, one for the client and one for the server process respectively. The client process handles all of the interaction with the user. For a terminal user this would normally be handling forms and all user I/O. The client code usually defines the start and end of global transactions. The server process handles and controls all access to the resource managers (eg Ingres). The server application code contains all of the SQL statements to access the databases.

The code invoked in a server process by a client is written in the form of a service routine. A server may contain many such service routines. The Name Server routes messages between application processes on one or more systems. Applications may thus be moved from one system to another simply by changing the contents of a configuration file - without altering any of the application code. This can be expanded to support the distribution of an application across multiple systems connected by a network. Data can be partitioned across multiple systems eg surnames A-M and N-Z. An identical Application Service can exist on each machine. The configuration file provides routing information that can direct a request to the appropriate system depending on the value in the data field. In modern environments it will be the enhanced client-server model that will typically be adopted.

## Three Tier Example

In Figure 3, the under-the-bonnet standard is XA which dictates standards-based communication between

# Visual Basic™ goes Client/Server!

NEW!

## Q+E® MultiLink•VB links Visual Basic to all major databases

With Pioneer Software's new Q+E MultiLink•VB, you can create powerful Visual Basic database applications that access data in over 20 PC and SQL database formats. Now you can build true client/server database applications and distribute them *Royalty Free.*

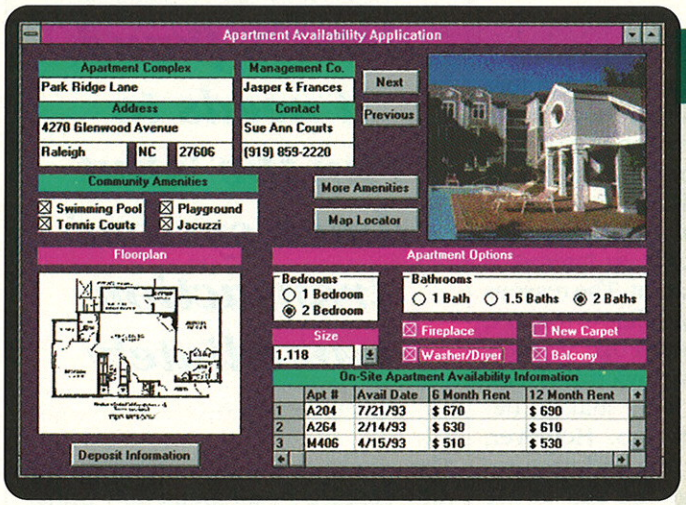
Q+E MultiLink•VB is the fastest way to build database applications—even if you need to access data from multiple formats. Use it to display data residing in several different databases together on the same form. Because Q+E MultiLink•VB works the same way for all database formats, you don't have to learn a different interface for each database. It's not only easy to learn—it's easy to use. You can even define sophisticated queries without knowing SQL.

### Create powerful database applications

Q+E MultiLink•VB includes several graphical development tools so you can quickly deliver powerful database applications. Take advantage of the database custom controls, the Database Manager, and the Query Builder to transform Visual Basic into a complete client/server development tool.

### Custom controls eliminate code

Custom controls added to the toolbar enable you to design your applications without writing code. In Q+E MultiLink•VB, you'll find the same controls included in our award-winning database for Visual Basic, Q+E Database/VB, and more.




Quickly create powerful Visual Basic database applications.

### Database management made simple

Use the Database Manager to create, view, modify, and delete databases from within Q+E MultiLink•VB. Design a database table and specify the field names, data types, and field lengths—then link the table to your application.

### Build sophisticated queries

Retrieve data from more than 20 database formats with our highly acclaimed Query Builder. The Query Builder leads you through the creation of SQL Select statements in a few simple steps. Just point and click to define the entire query before sending the request to the database.

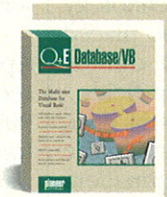
### Familiar with Q+E Database/VB?

New Q+E MultiLink•VB provides a seamless transition for Q+E Database/VB applications. In a matter of minutes, you can use Q+E MultiLink•VB to link applications built with our first Visual Basic product to all major databases. Call for details on our special offer.

### Multi-Database Access

Oracle • SQL Server
dBASE • INGRES
Paradox
Sybase • DB2
AS/400 (SQL/400)
Btrieve • OS/2 DBM
INFORMIX
NetWare SQL
Excel • PROGRESS
SQLBase • XDB
SQL/DS
Tandem NonStop SQL
Teradata
text files
HP ALLBASE/SQL
HP IMAGE/SQL

Gateways:
IBM DDCS/2
Micro Decisionware
Sybase Net-Gateway




### Need a database?

Pioneer Software's award-winning database for Visual Basic, Q+E Database/VB, is all you need to create network-ready Visual Basic database applications. It's dBASE compatible too! Try it today for only **£145**


To order your copy of Q&E MultiLink VB for **only**

**£325**

Telephone or Fax your order now on
**Tel (0273) 483979**
**Fax (0273) 486224**
Don't forget to add delivery £8.00 and VAT

CONTEMPORARY Software

1 Castle Ditch Lane, Lewes, East Sussex, BN7 1YJ Telephone (0273) 483979 Fax (0273) 486224
A Division of The Contemporary Computing Company Limited



CIRCLE NO. 076

the transaction manager and the Resource Manager (Ingres). What is far more relevant at the Application Client level is the way you call a service (such as Debit, New or Credit). The major standards defined here by X/Open are the XATMI and TX standards. These are to the client what XA is to the RDBMS.

## Role of Windows

Standish predicts that 80% of future OLTP projects will require some form of PC integration. New applications are likely to use a GUI interface. Migration of existing applications may require block-mode terminal emulation. The reason for this upsurge in interest is not immediately apparent. The reasons for GUI-TP are subtly different to those found for typical knowledge workers on a PC - but just as powerful. Traditional TP interfaces were initially developed many years ago. However, practices correct for the architectures of over a decade ago are still in use today, in totally different environments. The traditional system had a large number of terminals. Applications typically had many static forms. Priority was set on minimising interaction with the TP system to reduce network traffic.

Using GUIs, modern client technology allows much more data to be placed on the screen. This output can be dynamic - based on values in fields - and can include new classes of data, such as images. The client can also be used intelligently to cache information. The client is able to store not only the current form, but previous screens too. Application-specific online help can also be held at the client. Suddenly, practices of twenty years ago are not valid as they no longer require communication to the TP monitor across the network.

***When a bank detects fraud, it needs to run the 'shut down system' transaction immediately***

The power of this new architecture is fully enabled by adopting new principles to application design. The traditional application on a terminal behaved in a very linear fashion, with as much information crammed onto a form as possible. The power of the new architecture is enabled by using event-driven, object-oriented principles. Already, this change has brought with it major advantages.

Applications have been drastically reduced in size, with as much as a 90% reduction in the number of forms required. Since forms are dynamic they can change in context with what is being displayed. Visual feedback is provided and the context of forms is preserved ie if you enter a product code on one form and proceed to other forms where the product code is required, you do not have to go back to find out the code (and lose all of the subsequent work you have done). Instead the previous window can be referenced and the value copied and pasted to the other screen. In addition, the GUI offers a way to integrate between disparate systems. For instance it is possible to display all relevant information on a person, such as Current and Deposit account details.

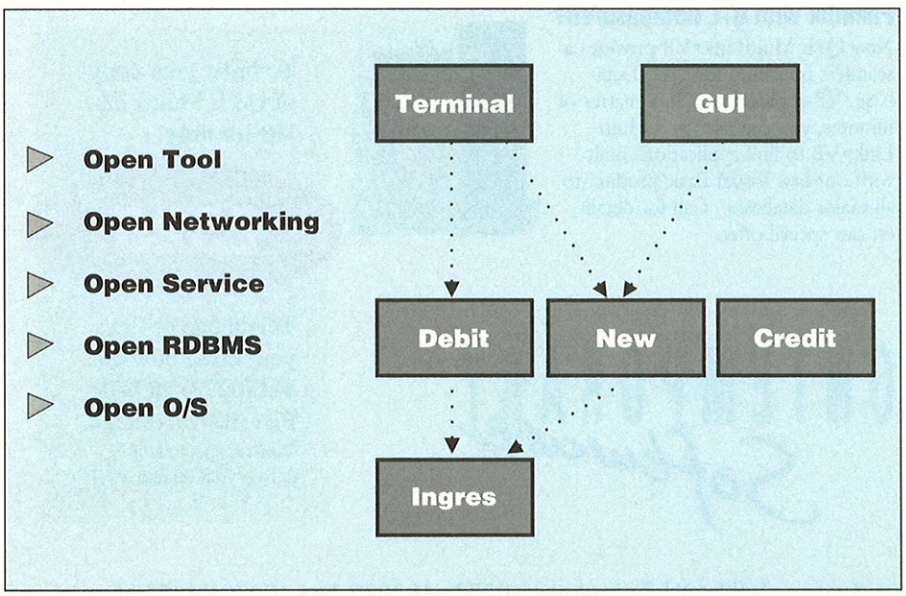



*Figure 3 - Open OLTP*

## TP Management

Systems management is seen as a key enabling technology for rightsizing. Vendors must provide improved 'people efficiencies' with combined system and network management products that address the 'client' desktop device as well as the distributed server device. TP systems are subsystems in their own right with resource demands and queues. They are ideal candidates for GUI monitoring and management. Through this approach, bottlenecks can far more easily be spotted and appropriate configuration made.

## TP Monitors

A number of Open Systems TP monitors exist. The most prominent are Tuxedo (Unix Systems Laboratories/Novell), Encina (Transarc), CICS/6000 (IBM) and Top End (NCR).

## Conclusion

Open Systems Distributed Transaction Processing is set to increase the use of the relational database system enormously. TP monitors coöperating effectively with the RDBMS will enable very high throughput, high availability systems to be built across distributed architectures. What is happening is, not one, but a number of technologies are converging together to bring forward large scale open systems distributed data servers.

EXE

---

*Ian Howells is the Product Marketing Manager for Ingres UK (0734 496000). In this role he has been the company representative on the OMG. He regularly speaks at public seminars and edits technical articles.*

# Fax Finale

*In the final part of his series on fax, Andrew Margolis outlines both the EIA Class 1 command set and the T.30 fax session protocol, and ties up a few loose ends.*

Last month, I described Class 2 fax modems. I will now end the series with Class 1. The complete EIA/TIA-578 Class 1 command set ought to appeal to minimalists. It is shown in Figure 1, and consists of commands to send or receive data in T.4 fax format and commands to send or receive data in an HDLC frame. These four data handling commands take a parameter which sets the speed and modulation scheme to use (for instance, `AT+FTM=96` tells the modem to prepare to send data at 9600 bps using V29 modulation). There are two other commands: one enables you to wait for a specific amount of time - mostly used when changing modulations, and one which waits for silence on the line - for error recovery. You also need to know that the command `AT+FCLASS=1` places the modem into the correct mode to send or receive faxes.

| | |
|---|---|
| +FTM= | Transmit Data |
| +FRM= | Receive Data |
| +FTH= | Transmit with HDLC Framing |
| +FRH= | Receive with HDLC Framing |

All the above commands take a parameter indicating the speed and modulation to use as follows :

3=v.21 ch 2 300 bps
24=v.27 ter 2400 bps
48=v.27 ter 4800 bps
72=v.29 7200 bps
96=v.29 9600 bps
145=v.17 14400 bps

Alternatively, the capabilities of the modem can be ascertained using a ? (e.g AT+FTH=?)

| | |
|---|---|
| +FTS= | Stop transmiting and wait |
| +FRS= | Detect Silence and wait |

Both the above commands take a parameter indicating the number of 10 ms intervals to wait

*Figure 1 - Basic Class 1 fax modem commands*

The easy part in sending or receiving a fax is squirting out or gobbling up the T.4 data. This is very similar to handling T.4 data with a class 2 fax modem, which we looked at last month. The difficult part is negotiating with the remote fax. All a class 1 fax modem provides you with is the ability to send and receive HDLC data frames. You have to do the work of composing and interpreting these frames yourself. To do this it is essential to know something about the CCITT T.30 recommendation, which describes the five separate and consecutive phases of a Group 3 facsimile transmission. Understanding the technical details of how this works is not only essential if you want to program Class 1 fax modems, but is also required for debugging other fax modem classes and for making sense of much technical documentation. Therefore, a digest of the important parts of the T.30 recommendation follows. As with all the information presented in this series, commercial developers are advised to look up the original.

## Phase A

Phase A handles the call set up. Initially, this is much the same as setting up an ordinary data call, and covers dial tone detection and phone number dialling at the calling station, together with ring detection and call answering at the called station. It only starts getting complicated when the two stations try to identify whether a call is a fax or not.

A calling station optionally identifies itself as a fax by sending a CNG (CallING) tone (1100 Hz in 500 ms pulses at three second intervals), which is the tone you have probably heard when a fax machine calls up on your voice line. During this time, it is also listening for a one second preamble from the answering station comprising a continuous stream of HDLC flag bits (01111110) sent at 300 bps. Once this is detected, the calling fax goes on to Phase B. An answering fax optionally identifies itself 2.15 seconds after answering with a 2100 Hz CED (CallED) tone sent for 3.3 seconds and then waits for 75 milliseconds before sending the preamble described above. (All timings +-15%).

Though the CNG and CED tones are optional, their main purpose is to enable human beings to tell that there's a machine at the other end of the line, and most fax systems implement them. However, one of the T.30 options for manual fax machines is to handle

| FIGURE | ADDRESS | CONTROL | FCF | DATA | CRC-16 | FLAG |
|---|---|---|---|---|---|---|
| 01111110 | 11111111 | 11001000 | 110000001 | 00000000<br>01110000<br>00010000 | 01001011<br>10111101 | 01111110 |
| | FF | 13 | 83 | 00<br>0E<br>08 | D2<br>BD | |

Bytes as seen by application

Bits actually received

*Figure 2 - Sample HDLC Frame*

Phase A entirely through an appropriate verbal exchange. So-called silent faxes take advantage of this option to skip CNG and CED tones completely even though this shouldn't ever be done for automatic fax machines.

Both phase A and the transition to phase B are handled automatically by a Class 1 fax modem as part of the ATD dial or ATA answer procedures.

## Phase B

Phase B covers the pre-message negotiations. This consists of an exchange of HDLC frames and begins at 300 bps, using a V21 modulation scheme. As a bare minimum, the answering station kicks off by following its preamble with a DIS frame (Digital Identification Signal) and the calling station responds with either a DCS frame (Digital Command Signal), if it wants to send a fax, or a DTC frame (Digital Transmit Command), if it wants to poll for a fax. These frames outline what the capabilities of the respective machines are, what options they support, and are used to establish the parameters which should be used for the rest of the session.

The basic DCS and DTC frames are optionally preceded by additional items of information, the most commonly implemented of which are the CSI frame (Called Subscriber Identification) from the answering station, and either a TSI frame (Transmitting Subscriber Identification) from the calling station, if it is sending a fax, or else a CIG frame (CallInG subscriber identification) if it is polling. These frames contains 20 character identity fields with the telephone number of each machine, and are used both for security in polled environments and also more generally to keep fax logs.

***There isn't much point in having a large buffer if it doesn't implement sensible flow control thresholds***

Subsequent negotiation procedures are controlled by the transmitting station, which initiates commands and waits for responses from the receiving station. The timings here are fairly important. The frames containing the commands or responses are never more than three seconds long, and if no reply to a command is received within three seconds, then it can be retransmitted up to three times. If there is still no reply, then the transmitting station sends a DCN frame (DisCoNnect) and terminates the session. The whole Phase B negotiation process must be completed inside 35 seconds.

As an example, assume that a fax is being sent, that the first DIS/DCS frames have been exchanged as described above, and that a data rate of 9600 bps has been agreed on. After the transmitter has sent, a DCS it delays for 75 ms (a standard wait before any modulation change) and begins a V.29 9600 bps training sequence of 0s for 1.5 seconds, referred to as the TCF (Training Check Frame). This training sequence effectively enables the modem to work out whether the line is of good enough quality to communicate with the chosen speed. After the TCF, the transmitter returns to the 300 bps data rate to await a response. If the receiver is successfully trained at 9600 bps, it sends a CFR frame (Confirmation to Receive): the Phase B negotiations have been successfully completed.

If the receiver failed to recognise a 9600 bps TCF for 1.5 seconds, it would respond with a 300 bps FTT frame (Failure To Train). In this event, the transmitter would send another DCS frame with a fallback data rate, followed by a training sequence at the lower speed - for instance, FTT at 9600 bps would prompt an attempt at 7200 bps, followed by 4800 bps and then 2400 bps if still unsuccessful. Other possible error responses would generally require re-sending the DCS frame and include CRP (Command RePeat), indicating that the DCS didn't make sense, or another another DTC or DIS, indicating that no proper DCS frame had been received at all. The same action could reasonably be taken if no response at all was received.

Where the first bit of the FCF is listed as an x this should be set to 1 if we received the DIS or reset to 0.

| | | |
|---|---|---|
| DIS | answering capability follows | 00000001 |
| CSI | answering station identity follows | 00000010 |
| NSF | answering nonstandard facilities | 00000100 |
| DTC | polling capability follows | 10000001 |
| CIG | polling station identity follows | 10000010 |
| NSC | polling nonstandard facilities | 10000100 |
| DCS | transmitter parameters follows | x1000001 |
| TSI | transmitter station identity follows | x1000010 |
| NSS | transmitter nonstandard facilities | x1000100 |
| CFR | confirmation to received | x0100001 |
| FTT | failure to train | x0100010 |
| EOM | end of document | x1110001 |
| MP | Send of page | x1110010 |
| EOP | end of transmission | x1110100 |
| MCF | message confirmation | x0110001 |
| RTP | message confirmation with retrain | x0110011 |
| RTN | message not received with retrain | x0110010 |
| DCN | disconnect | x1011111 |
| CRP | command repeat | x1011000 |

*Figure 3 - Main T.30 HDLC Facsimile Control Fields*

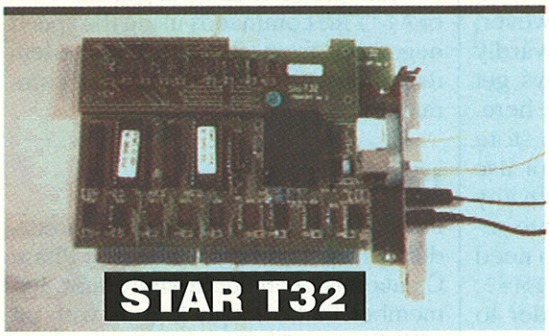
## Fax Session Parameters

To send or receive V21 HDLC frames via a Class 1 fax modem, simply issue the AT+FRH=3 or AT+FTH=3 command and wait for the CONNECT response. (The only exception to this is when dialling: this implies an AT+FRH=3 as soon as the preamble is seen). The modem will take care of the flag bytes and the frame check sequence bytes (FCS): these are inserted when frames are transmitted; the flags

# VOICE PROCESSING

# INTERACTIVE VOICE RESPONSE

# COMPUTER INTEGRATED TELEPHONY




## THE DEVELOPER'S CHOICE FOR SINGLE LINE TELEPHONY APPLICATIONS ON THE PC *

**Star T32 - a half length card for ISA Bus IBM compatible PCs**

- WINDOWS DRIVER FOR WINDOWS V3.1
- TSR TOOLKIT FOR MS-DOS AND OS2 V2.x
- WINDOWS BASED SOUND EDITOR
- SCRIPT LANGUAGE FOR EASY PROTOTYPING
- MICROPHONE AND LOUDSPEAKER
- DTMF RECOGNITION AND DIAL OUT
- PULSE DIAL OUT
- 16 BIT DIGITAL SOUND QUALITY
- SOCKET FOR LOCAL PHONE
- RECORDING FROM LINE, MICROPHONE OR LOCAL PHONE
- PLAYBACK TO LINE, LOUDSPEAKER OR LOCAL PHONE
- TIME BREAK AND EARTH RECALL
- OFF HOOK SENSORS FOR BOTH CARD AND LOCAL PHONE
- JACK FOR ALARM EVENTS
- SOFTWARE READABLE SERIAL NUMBER


Staria has been developing single line telephony cards since 1988. Our customers have produced applications in voice messaging, message broadcasting, telemarketing, plant monitoring, computer monitoring, telephone servicing, network testing, caller conferencing etc. Now, with added windows capability, the T32 is available for personal desktop use.

**COMING SOON:**

- SPEAKER INDEPENDENT, CONNECTED SPEECH RECOGNITION.
- DIAL OUT FROM OTHER WINDOWS APPLICATIONS.
- FAX/MODEM SIMULATION.
- SUPPORT FOR .WAV FILES

**PRICES:**

- DEVELOPER'S KIT: **£600.00** (Includes one card)
- ADDITIONAL SINGLE CARDS: **£300.00** QUANTITY DISCOUNTS

IVR for the PC
Staria

* ADDITIONAL CARDS CAN BE ADDED FOR UP TO 32 LINES

**Staria Limited** (Ask for Brian Atherton)
**TEL: 0989 768 687 FAX: 0989 768 980**
13 Market Place, Ross-on-Wye, Herefordshire HR9 5NU

are deleted and the FCS is checked when a frame is received. If a frame is in error, the modem reports ERROR, else it reports OK. However, the programmer has to check bit 5 of the control field for each frame received. If it is 0, another frame is due to be received, but if it is 1, then the frame is a final one.

Figure 2 shows a sample HDLC frame from a fax session. Note that all faxes default to using address 11111111. Also, bit 5 of the HDLC control field is set to 1 to indicate that the frame is a final one in the procedure, as described above. The data in the example is a DCS frame sent by a transmitting caller about to fall back to 7200 bps after a failure to train as described above. Other possible frames have different FCF fields - the main ones are shown in Figure 3. Most frames don't contain any data, in which case the FCF field conveys all the information required. One obvious exception are the frames such as the TSI, containing ID information. However, the most important frames with data in are the DIS/DTC/DCS frames, such as the one shown. All these must contain at least three octets of data.

| | |
|---|---|
| 1 | 0 = Group 3 |
| 2 | 0 = Group 3 |
| 3 | 0 = Group 3 |
| 4 | 0 = Group 3 |
| 5 | 0 = Group 3 |
| 6 | reserved |
| 7 | reserved |
| 8 | reserved |
| 9 | 1 = transmitter T4 |
| 10 | 1 = receiver T4 |
| 11,12,13,14 | data rate (see Figure 5) |
| 15 | 1 = fine resolution |
| 16 | 0 = standard 1-D coding |
| 17,18 | 0,0 = 1728 pixels wide |
| 19,20 | 0,0 = A4 length |
| | 0,1 = unlimited length |
| 21,22,23 | minimum scan line time |
| | 000 = 20 ms |
| | 100 = 40 ms |
| | 010 = 10 ms |
| | 001 = 5 ms |
| | 110 = 10 ms (5 ms fine) |
| | 011 = 20 ms (10 ms fine) |
| | 101 = 40 ms (20 ms fine) |
| | 111 = 0 ms |
| 24 | extend bit |

*Figure 4 - DIS/DTC/DCS bit definitions*

Figures 4 and 5 detail the meaning of each bit of these three octets. There are extra bit fields possible, which can be flagged by setting the extend bit (24). The latest revision of T.30 now defines 65 bit fields. However, all Group 3 faxes are downwardly compatible and you can always get by with just the 24 bits shown here. Applications software needs to store the parameters from phase B for use in subsequent phases, along with any additional pages of a fax. Before receiving a fax, programmers also need to issue AT+FRM=?, or AT+FTM=?, before transmitting one, in order to determine the capabilities of a modem. (A number of chipsets can send faxes at V.29 9600/7200 bps but can only receive at V.27 4800/2400 bps.)

## Phase C

Phase C is the message transmission phase and begins once a TCF/CFR exchange has been successfully completed. It is a unidirectional phase - the transmitter returns to the negotiated high speed data rate, sends another training sequence, and then sends the fax data in encoded T.4 format (assuming that was what had been negotiated). The only slight departure from the T.4 sending method we described last month (using the DLE ETX pair to mark the end of a page and a DLE-DLE pair to send one DLE) is that each line of pels in the fax will usually need padding out with fill bits of 0 before the EOL, in order that a line doesn't drop below the minimum scan time negotiated. (Bear in mind that the T.30 specification assumes that the receiver is a fax machine which is printing out data as it receives it, and that if lines arrive too fast then they don't get printed.)

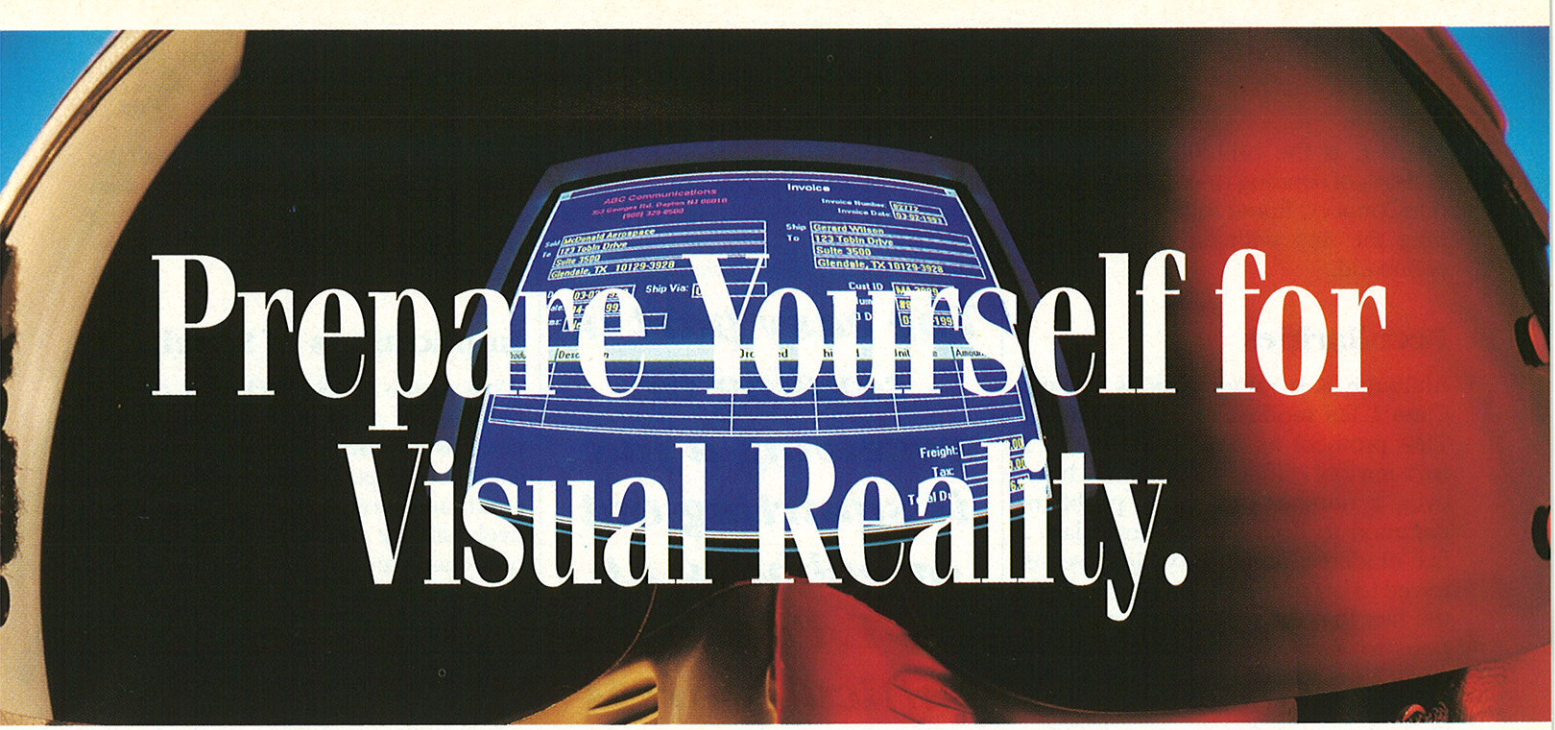
When using a class 1 fax modem, it is necessary to switch from low HDLC negotiating speed to a higher data speed to send data (or to train). This needs to be prefixed with a 75 ms delay, which can be achieved by issuing the command AT+FTS=8 and waiting for an OK. It will then be safe to enter phase C data via the AT+FTM or AT+FRM commands using the speed negotiated from phase B. The modem responds with CONNECT to these commands.

## Phases D & E

Phase D is the post-message procedure phase and begins after the Phase C data has been successfully sent. Remember that, though it ends with an RTC (Return To Control) sequence of six EOLS, this is signified by the DLE ETX sequence for a fax modem. The transmitter and receiver return to 300 bps and the transmitter then sends a post-message frame. There are a number of these, but the two most common are an MPS (MultiPage Signal), indicating that, though one page has ended, the receiver should return to Phase C for the next page, or else an EOP (End Of Procedure) indicating that the fax and session are now complete and that the disconnect phase should begin. The receiver responds with an MCF frame (Message ConFirmation) if all is well. Other possible responses are PIP (Procedural Interrupt Positive) indicating that, though the page was received, further transmissions are not possible and RTP (ReTrain Positive) is required. This indicates that the page was received but that phase B should be reëntered as a new TCF or DCS. Negative versions of the last two messages are also possible (PIN and RTN) if the page wasn't received. But there isn't a lot that can be done about this; the usual action taken is to disconnect and try again later.

| | DIS/DTC capability | DCS rate |
|---|---|---|
| 0000 | V.27ter fallback only | 2400 V.27ter |
| 0100 | V.27ter only | 4800 V.27ter |
| 1000 | V.29 only | 9600 V.29 |
| 1100 | V.27ter V.29 | 7200 V.29 |
| 0001 | | 14400 V.17 |
| 0101 | | 12000 V.17 |
| 1001 | | 9600 V.17 |
| 1101 | V.27ter V.29 V17 | 7200 V.17 |

*Figure 5 - Data Rate definitions for bits 11, 12, 13, 14 in*




# Prepare Yourself for Visual Reality.

## Introducing ProtoGen+. Visual tools with the most awesome workbench ever created for Windows development.

The future has arrived—a complete point-and-click, WYSIWYG workbench that lets you create dazzling applications without writing a line of code.

**Discover the ease and productivity of visual development!**

- Visually develop screens
- Create a menu and connect screens & dialogs
- Test the application's screen flow in a live environment
- Instantly generate source code in Pascal, C or C++

Paint your screens. Design a menu and link screens together. Test the flow in a live environment, and generate code for ANSI C, C++ for OWL or MFC or Pascal with objects. It's that easy.

And this open! ProtoGen+ will work with your database, compiler, libraries and extensions. Powerful add-on features, like SQLView offer workbench access to multiple databases to develop client/server and xBase applications. Snap-in components make ProtoGen+ open to future development technologies—whatever they may be.

ProtoGen+ is easy to learn, speeds your development cycle and protects your investment by generating C,C++ and Pascal. We guarantee you'll prototype and generate exciting applications faster than you ever thought possible!

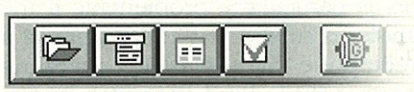

Point-and-click to paint screens with bitmaps, icons, tables, data validation, custom colors, fonts, 3D effects, visual toolbars, status lines, balloon help, MDI and more!

**The most powerful, open set of Visual Development Tools ever!**

| | | | |
|---|---|---|---|
| Build windows, forms, dialog boxes, tool bars | **Dialog Editor** | **Menu Designer** | Quickly create a menu using templates |
| Output C, C++ and Pascal with Objects | **Code Generator** | **ProtoView Screen Manager** | Data validation, 3-D effects, MDI and more |
| Create new designers! Source included | **Custom Visual Designer** | **Win-Control Library** | Rich library of visual control objects |

**Snap-in Code Generators**

License our technology to create new code generators



Bring your applications to life using the latest visual design tools!

**Fasten your seatbelt for ProtoGen+!**

**only £149 (list price £299)**

**Tel: 0273 483979**

**Fax: 0273 486224**

**ProtoGen+'s SQLView access to multiple databases is available at an additional price; ask about it when you call.**

MONEY BACK 30 DAY GUARANTEE

CONTEMPORARY Software

1 CASTLE DITCH LANE, LEWES, EAST SUSSEX, BN7 1JY
TEL: 0273 483979 FAX: 0273 486224

➤ CIRCLE NO. 082

Phase E is the call release phase. It consists of sending or receiving a DCN (DisConnect) frame followed by dropping the line with an ordinary ATH command.

## Peculiarities

Although it is rather tedious using a Class 1 fax modem to implement the T.30 protocol as described above, it isn't terribly difficult. There are, however, a number of general problems common to both Class 1 and Class 2 fax modems, some of which we referred to last month, which derive from the fact that they all convert synchronous to asynchronous data and *vice versa.*

Since Group 3 fax communications, most commonly occur synchronously at 9600 bps, the asynchronous link has to be able to run at least at 12000 bps if it is to be able to keep up with the 120 characters per second that the synchronous link will deliver in T.4 mode. The specifications simply state that the computer-to-modem speed must be at least 25% faster than the maximum expected line connection speed. As 12000 bps is not a usual option for serial ports, there's a *defacto* standard for both Class 1 and Class 2 modems, which requires that the computer serial port be driven at a fixed speed of 19200 bps. This appears to be a universal extension to the specifications and is mandatory no matter what speed the fax is being sent or received.

***The easy part in sending or receiving a fax is squirting out or gobbling up the T.4 data: the difficult part is negotiating with the remote fax***

All fax modem service class specifications insist that at least XON/XOFF flow control must be available to stop the modem buffer overflowing when sending a fax. It is assumed, when receiving, that the computer will be able to take anything that the modem chucks at it, so there is no handshaking or flow control available when receiving a fax.

## Implications of Synch

What makes XON/XOFF flow control particularly difficult is that synchronous data must be sent as a continual stream of bits. Unlike asynchronous communication, where there is effectively silence between the stop bit of one character and the start bit of the next, there is no such thing as silence on a synchronous link, as the receiver samples bits at specific intervals as either 0s or 1s. In order to maintain continuous communications, all fax modems have buffers which are needed to handle speed conversion from 19200 as well as to supply a continuous stream of bits to send.

Obviously, a large buffer is useful if you are sending a fax from disk: being able to dump a whole sector into the modem buffer makes the time constraints involved in accessing the disk a little less critical. Given the size of T.4 fax images, storing them in memory is not always practical. However, to avoid the problem of synchronous underrun (having no data to transmit), the size of the buffer is of less importance than the way it works with flow control. There isn't much point in having a large buffer if it doesn't implement sensible flow control thresholds. An example will make this clearer.

The Class 2 command AT+FBUF? is supposed to return the total buffer size, XOFF threshold, XON threshold, and the number of bytes currently in the buffer. There is no comparable class 1 command, but for both types of modem, buffer sizes as low as 104 bytes with an XON threshold of 16 bytes are not uncommon. If a fax is being send at 9600 baud, the 16 bytes left in such a buffer will be emptied in only 10 ms. The largest single cause of fax transmission failure using a fax modem may well be the inability of some serial port handlers to respond to an XON from a modem within the period of time that it takes the rest of the buffer to empty. This is particularly problematic if the handler is subject to the demands of a preëmptive multi-tasking operating system which *can't* guarantee access to a CPU at least once every 10 ms. The only practical solution to this problem is either to buy a fax modem with a

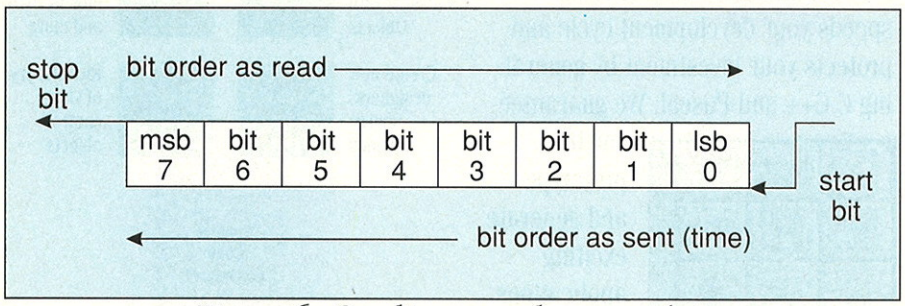



*Figure 6 - Sending asynchronous data*

```
unsigned char reverse (unsigned char byte)
{
static unsigned char backbyte, newbit, mask ;
for (backbyte=0,newbit=0x1,mask=0x80 ; mask ; newbit<<=1,mask>>=1)
if (byte & mask) backbyte |= newbit ;
return (backbyte) ;
}
```

*Figure 7 -*
*C function to reverse the order of the bits in an 8-bit byte*

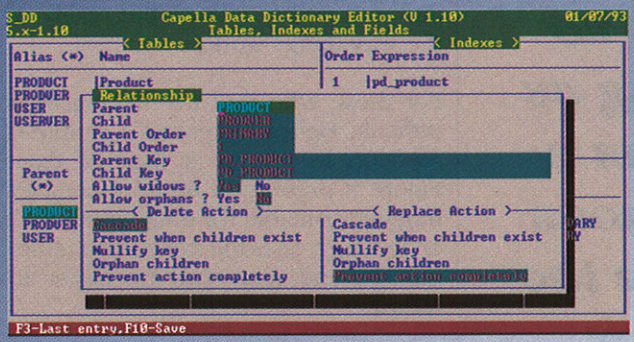
```
        mov     ah,al
        mov     cx,8
reverse:
        shl     ah,1        ; shift bit out into carry from left
        rcr     al,1        ; shift bit from carry in to the right
        loop    reverse
```

*Figure 8 - 80x86 assembler fragment to reverse a byte in al*

# Capella

## Professional Development System

*"A Combined Command, Function and Class Library and Development Surface"*




**The Capella Professional Development System (Capella PDS) is the complete application development solution for Clipper 5 developers.**

## FEATURES

- **INTEGRATED ACTIVE DATA DICTIONARY with full Referential Integrity**
- **DATABASE AND TABLE CLASSES -including tables for DBFNTX, arrays, SIx Driver, Mach SIx, Comix Driver, Clipmore, Flexfile II RDD, dBxStore II and the RDD Kit**
- **TABLE AND MENU SECURITY CONTROL - two independent and cooperative security systems manage access to menu options and access to data.**
- **BACKGROUND TASKS - built in support for almost any part of CA-Clipper**
- **USER INTERFACE - full user interface primitives, composites and metafunctions**
- **GET SYSTEM -Fully integrated get system with numerous enchancements**
- **POWERFUL PRINTING, QUERY BY EXAMPLE, REPORTING AND AD HOC REPORT WRITER**

The Capella PDS is a Library for all programmers by offering five programming levels in one product. Programmers can pick and choose between any or all of the developement surface, commands, high level function, low level functions and classes. For Example:

| Command | High Level Function | Modal Function | Class |
|---|---|---|---|
| S_BROWSECMD | s_BrowseNew() | s_BrowseModal() | s_View() |
| S_LOOKUPCMD | s_LookUpNew() | s_LookupModal() | s_View() |
| S_REPORTCMD | s_ReportNew() | s_ReportModal() | s_Report() |
| S_MENUCMD | s_MenuNew() | s_MenuModal() | s_Menu() |
| S_QBECMD | s_QBENew() | s_QBEModal() | s_QBE() |
| S_ALERTCMD | s_AlertNew() | s_AlertModal() | s_Alert() |
| S_BOXCMD | s_BoxNew() | n/a | s_Box() |

Capella Software have just announced that two parts of the Capella PDS - dbClass and the Capella DDE- will now also be available as independent products

## dbClass

NOW AVAILABLE

dbClass a "plug and play" class library for Clipper 5.01, 5.01a and 5.2x can be used along side most Clipper code or as a complete replacement for all RDD based or API based data access. dbClass classes range from database, table, index, record to relationship and RDD classes and owners of Class(y) and SuperClass can inherit from dbClass classes to make their own enhancements.
Also support for: *Referential Integrity (through any depth of relationship) for both delete and replace *Triggers which can be set at field, table or database level...and much more

***"dbClass is the best way to achieve RDD and API independence", Rick Spence "The only way to futureproof your Clipper Applications today", Terry Brown, Clipper Consultant***

## The Capella DDE

NOW AVAILABLE

The Capella Data Dictonary Editor (DDE) is a stand alone Data Dictionary Integrated Developement Environment (IDE) for use with CA-Clipper applications. The Capella DDE allows the definition of a complete database via an easy to use IDE and its ability to load the data dictionary from exsiting data files makes it easy to use immediately. As it updates data files with changes when they are made the application and its data files are never mismatched. Referential Integrity rules for cascade, restrict, nullify, orphan and prevent can be defined for both delete and replace operations through any number of levels of relations.
Also included is PROTOTYPE.EXE, a basic prototype program which reads the data dictionary and provides a simple application based on its contents.

***"Anyone who is not using a product like the Capella DDE is wasting a lot of effort producing programs that are less robust than they should be.", Steve Hallam, Bristol Clipper User Group***


QBS

QBS Software Ltd

10 Barley Mow Passage, London W4 4PH

Tel :081 994 4842 Fax :081 994 3441

BBS : 081 747 1979

large buffer and sensible thresholds, or else to implement interrupt-driven transmission routines which themselves incorporate a large enough buffer to keep the transmission going.

## Backwards with Async

A more subtle problem is caused by the way that data is transmitted. Given that fax transmission is bit-oriented rather than byte-oriented, it makes sense for the first bit received by a fax modem from a computer to be the first bit which it sends to the remote machine, and for the first bit received from a remote machine to be the first bit sent back to the computer. But asynchronous data has its own way of working, which is that the least significant bit is always transmitted first, and the most significant bit, last. This is illustrated in Figure 6, where it is apparent that the bits in a byte are actually sent in the reverse of the order in which we naturally would read. We can see the problem this causes for a synchronous link when we attempt to transmit a byte aligned T.4 EOL pattern to a fax modem. The bytes read as follows:

```
00000000 00000001
```

The first byte consists of eight zero bits. Obviously it makes no difference which way round this one goes. However, reference to Figure 6 makes it apparent that the second byte will be transmitted asynchronously with the least significant bit 1 as the first bit sent. So when this bit is sent on first by the fax modem, it won't be sending an EOL at all. The same problem occurs when receiving data. If you refer back to Figure 1, it can be seen that the data bytes received by an application program are actually the reverse of the octets of bits received by the modem.

***There is no such thing as silence on a synchronous link, since the receiver samples bits at specific intervals as either 0s or 1s***

One solution to this problem is to program T.4 and T.30 data backwards. While for T.4 data this is simply counter-intuitive, for the T.30 HDLC frames it adds an extra level of complexity to any code and can introduce an extra source of bugs. Another solution, which we adopted for Class 2 modems last month, is to use the `+FBOR` command to tell the modem to reverse the bits for us: however, this isn't available for Class 1 modems.

Though there's a small performance penalty, my preferred method is to reverse the bytes on sending and on receiving. The extra time needed is the only disadvantage, but this is not critical. It is outweighed by the fact that a program can use a natural representation rather than a reversed bit-order. It is also the most logical thing to do, as it simply reflects the reality of the way things work.

For those who find bit manipulation difficult, Figure 7 presents a simple C function to reverse a byte of data. Figure 8 present an 80x86 assembler fragment suitable for embedding in C code which does the same thing rather quicker and may well be easier to follow.

There are two exceptions to the need to reverse fax data. One exception is when receiving with a Class 2 fax modems, as they always reverse received T.4 data before passing it on, even when `FBOR=0`. It seems that this was originally a bug in the Rockwell chipset, which everyone else copied, and consequently it never got fixed.

The other exception is that all fax IDs passed as data in T.30 and Class 1 CSI/CIG/TSI frames is already reversed: and not only is the bit order reversed, but the byte order is also (so 123-4567 is send as 7654-321). There may be a certain mad logic to this, but what it might be escapes me for the moment. However, it does finish the series on an appropriately messy note. Happy faxing !

EXE

---

*Andrew Margolis runs a software house and consultancy specialising in communications, emulations, fax, EPOS and utilities for a wide range of systems. He can be contacted at Margolis & Co on 081 8897755 and is always looking for work. While binaries for his fax software are available as shareware (see* `exe/files` *on* cix*), if you don't feel like writing up the source code all by yourself, you can obtain it from him when you register the shareware for £30. Though certain component level I/O routines are in assembler, you get all the C code for encoding, decoding, sending and receiving faxes together with all the relevant data structures. Considering the typing time you'd save this represents a* **real** *bargain.*

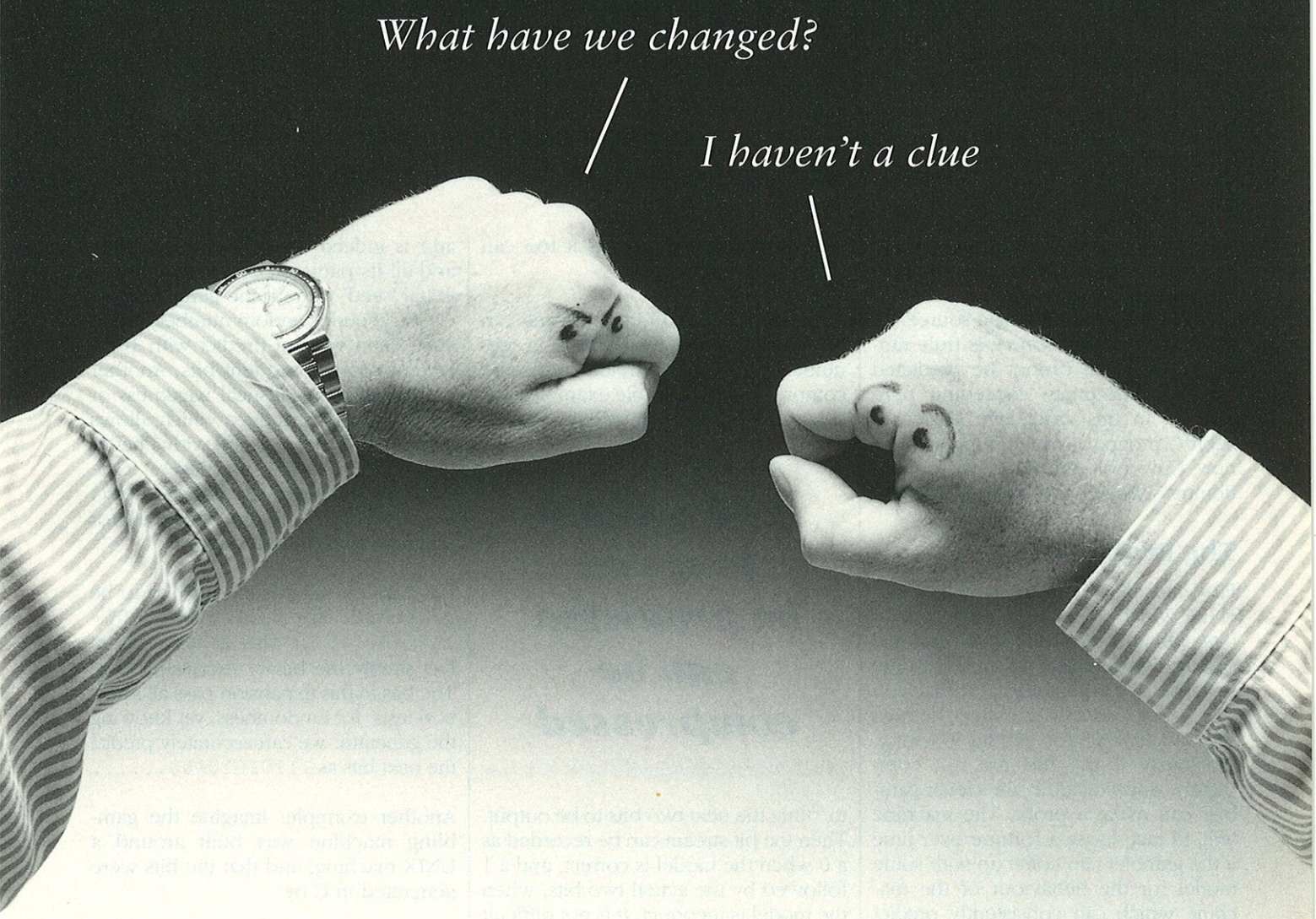
### Finding more facts on fax

CCITT Fascicle VII.3 contains the T.30 recommendation. This is one of the CCITT Blue Books which can be found in a few good technical libraries. I obtained my copy from the Science Reference Library in Southampton Buildings, Holborn, London. This establishment has not yet been privatised: admission is *still* free, though it charges for photocopies. It doesn't yet have the latest T.30 revision, which is published as CCITT White Document COM VIII-164.

There appears to be no UK library in the country with the EIA/TIA-578 Class 1 standard. However, either BSI (0908 221166) or Omnicom (0438 742424) will sell you a copy - possibly bought in from Global Engineering Documents in California (fax 0101 714 2617892) who appears to be the most widely used transatlantic source.

The only public domain account of Class 1 is the one compiled by Supra, which is to be found on its US bulletin board (0101 503 9672444) or from various Internet sites. The only technical support I've had from British modem manufacturers for Class 1 has been from Sonix (0285 641651). Developers seeking to purchase modems with adequate technical backup should **always** check whether source documents are available.

# Does your left hand know what the right is doing?




Get your hands on Intasoft's Configuration Management tools and it will!

**SMS** is our full-featured version control package: perfect for preventing users overwriting each others' changes and for automating the build process – as well as telling you *Who?*, *What?*, *When?* and *Why?*

Full configuration and change management can make anyone all fingers and thumbs. Not with **AllChange**, Intasoft's brilliant package which provides facilities for product design, baselines, life-cycles, user roles, change requests, configuration build and release management.

Our products will help you with standards (e.g. ISO9001, BS5750) and are backed by our outstanding support.

With all of this available for MS-DOS, Windows and UNIX at prices from as little as £490, we think you'll agree Intasoft are handy people to know!

**INTASOFT, Tresco House, 153 Sweetbrier Lane, Exeter, EX1 3DG.**
**Tel 0392 217670 Fax 0392 437877**

**NEW** *SMS Version 4 for Windows Now available*
**NEW** *AllChange – Change control & configuration management system*

INTASOFT

*Quality tools for professional software developers*

# How random is your generator?

*Is there any underlying pattern to that bit stream? Is that file compressible? Michael Scott presents a universal random bit tester.*

What do the successful gambler, wiretapper and data compressor have in common? The answer is the ability to predict the output of a data source. Of course, if the data source is truly random, its output cannot be predicted with an accuracy exceeding pure chance. In this article we present a short C program which implements a new Universal statistical test for randomness, developed by Ueli Maurer.

## The Problem

First consider a simple gambling machine which generates a sequence of bits emanating from some internal physically-protected source. Assume that you are a gambler, and that you win £5 for correctly guessing the next bit, and lose £5 for getting it wrong. Obviously if the bits are not completely unpredictable, the clever gambler can make a profit. The machine will, in fact, loose a fortune over time if the gambler can come up with some model for the behaviour of the machine which can consistently predict the next bit for even slightly better than 50% of the time. Of course the same applies to other more elaborate gambling activities, such as betting on horses, or playing the Stock Market.

Two individuals are attempting to communicate a message by phone using the unbreakable one-time-pad method of encryption. Each bit transmitted is Exclusive-ored with the next bit generated from a random source. At the other end, an identical random source generates identical bits which are XORed with the incoming bits, restoring the original message. A wiretapper is listening in. If the wiretapper can accurately predict the next bits from the 'random' source, it too can decrypt the message.

A bit stream that can be predicted can be compressed. Conversely a genuinely random bit stream cannot be compressed. As a simple example consider a model that can predict 75% of the time the next two bits to be output. Then the bit stream can be recorded as a 0 when the model is correct, and a 1 followed by the actual two bits, when the model is incorrect. It is not difficult to see that this simple recording scheme reduces the length of the bit stream by 25% on average.

***A bit stream that can be predicted can be compressed***

Are the gambler/wiretapper/data compressor wasting their time? They are if the bits they are examining are unpredictable, that is, if they emanate from a truly random source. A simple test is needed which will provide a strong indication as to the predictability or otherwise of the bits.

## How to Predict

There are a number of ways to predict the next bit(s). One is to discover the bit-stream generator (if such a generator is indeed the source of the bits) and all its parameters, for example its initial 'seed' value if the generator is a classic pseudo-random number generator. Then we can predict with 100% certainty all subsequent bits. Another is to identify a statistical weakness or pattern in the sequence which allows its output to be predicted with a better than 50-50 chance of success. One easy way to do this would be to detect the onset of cycling, that is to look for a repetition of the bit pattern.

For example what is the next bit to be generated after `110010010000-111110.....`? This generator is in fact simply the binary expansion of π! The bits in this expansion pass all statistical tests for randomness, yet knowing the generator we can accurately predict the next bits as `11101010100.....`

Another example: imagine the gambling machine was built around a UNIX machine, and that the bits were generated in C by

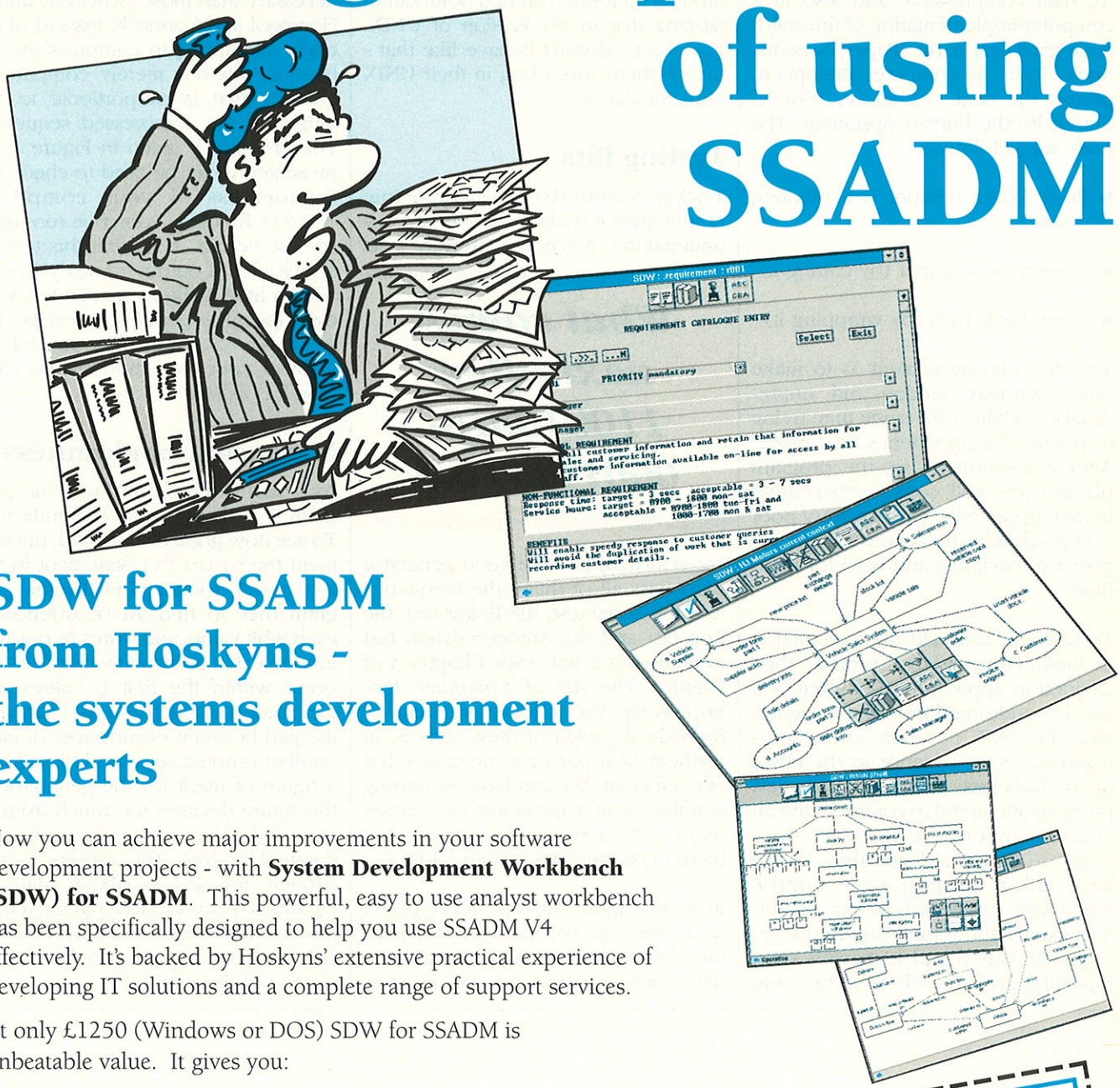
```
b=rand()&01
```

using the built-in pseudo-random `rand()` function. In this case even the most inebriated punter would do rather well - the sequence would be `....0101010101010101010101 0101010....` Using a PC, a rather more alert and patient gambler would also win eventually - for all PC compilers tested (Borland, Microsoft, Watcom) the bit sequence repeats after only 13,1072 bits.

## Children's Games

In 1986 Horspool & Cormack presented a paper at the Nineteenth

# Taking the pain out of using SSADM




## SDW for SSADM from Hoskyns - the systems development experts

Now you can achieve major improvements in your software development projects - with **System Development Workbench (SDW) for SSADM**. This powerful, easy to use analyst workbench has been specifically designed to help you use SSADM V4 effectively. It's backed by Hoskyns' extensive practical experience of developing IT solutions and a complete range of support services.

At only £1250 (Windows or DOS) SDW for SSADM is unbeatable value. It gives you:

- heavyweight support for SSADM V4
- an intuitive mouse driven interface
- a central data repository for true team working
- user controlled consistency and validity checking

Implementing and using SSADM need not be a headache. Hoskyns can help by providing not just the tools but also SSADM experts to support, train and advise. So if you're looking for a quicker, more effective way to use SSADM, return the coupon, or call us on 071-434 8111 today.

HOSKYNS
CAP GEMINI SOGETI

☐ I would like more information on SDW for SSADM.

☐ Please contact me to discuss my SSADM requirements.

Name

Job Title

Organisation

Address

Post Code Tel. No.

Return to: Project Systems Division FREEPOST SW5779. Hoskyns Group plc, Hoskyns South Bank, 95 Wandsworth Road, London SW8 2HG. A066

➤ CIRCLE NO. 085

Annual Hawaii International Conference on Systems Sciences, describing a Markov Modelling technique for prediction. They used it successfully for data compression, and also in a computer implementation of the children's gambling game 'Paper, Scissors, Rock'. In it, the computer attempts to predict the next symbol (P, S or R) output by the human opponent. The rules are as follows:

- Rock beats scissors (by blunting them)
- Scissors beats paper (by cutting it)
- Paper beats rock (by wrapping it)

The best playing strategy is to make one's own plays random and unpredictable, while at the same time trying to predict the opponent's next move. After a learning phase, the program played quite well, consistently beating human opponents, who are very poor at unpredictable behaviour. Patterns of play are inevitable and hard for us to hide.

To test and calibrate their prediction method, Horspool & Cormack then decided to apply it to the output of a good pseudo-random number generator, the UNIX `random()` function - regarded as far superior to the absolutely hopeless `rand()`. Their test program attempted to compress the bit stream generated from `b=random()&01`; an attempt that should have failed. Imagine their surprise when they succeeded! In fact, on further examination, the alleged random sequence was found to have a short repeating cycle of only 9198 bits, and was non-random even within that period. Horspool and Cormack had accidentally discovered a new application for their prediction method - as a random bit tester. I should point out in passing that in *my* version of UNIX, `random()` doesn't behave like that - maybe there was a bug in their UNIX implementation.

## Testing Bits

Testing a random bit generator for true randomness is traditionally not an easy undertaking. A whole battery of tests

***What comes next after 110010010-000111110..?***

are available, and the good generator must pass all of them: the Frequency test, the Serial test, the Poker test, the Run test and the Autocorrelation test to name but a few. (See Chapter 3 of Knuth's *The Art of Computer Programming*, Vol 2 for more details.) Individually, each of these tests is, in mathematical parlance, necessary but *not* sufficient. We can have increasing confidence in a generator that passes more and more of these tests - but there must remain a nagging doubt.

Recently Maurer [2] has developed a 'Universal' test, based on the Horspool and Cormack idea, which is able to detect any one of a very general class of statistical defects, including all those found by the five tests I have named above. Loosely speaking this test, while not in itself sufficient, is more necessary than most. However, unlike Horspool and Cormack, instead of actually attempting to compress the sequence, this test merely computes a quantity that is proportional to the length of the compressed sequence. The program is given in Figure 1. As presented it can be used to check the randomness of your compiler's `rand()` function - don't be too upset by the verdict. Note that this test requires a lot of output for analysis, so it takes a little while to run. To test your own favourite generator, rewrite the `generator()` function to deliver eight of your own random bits each time it is called.

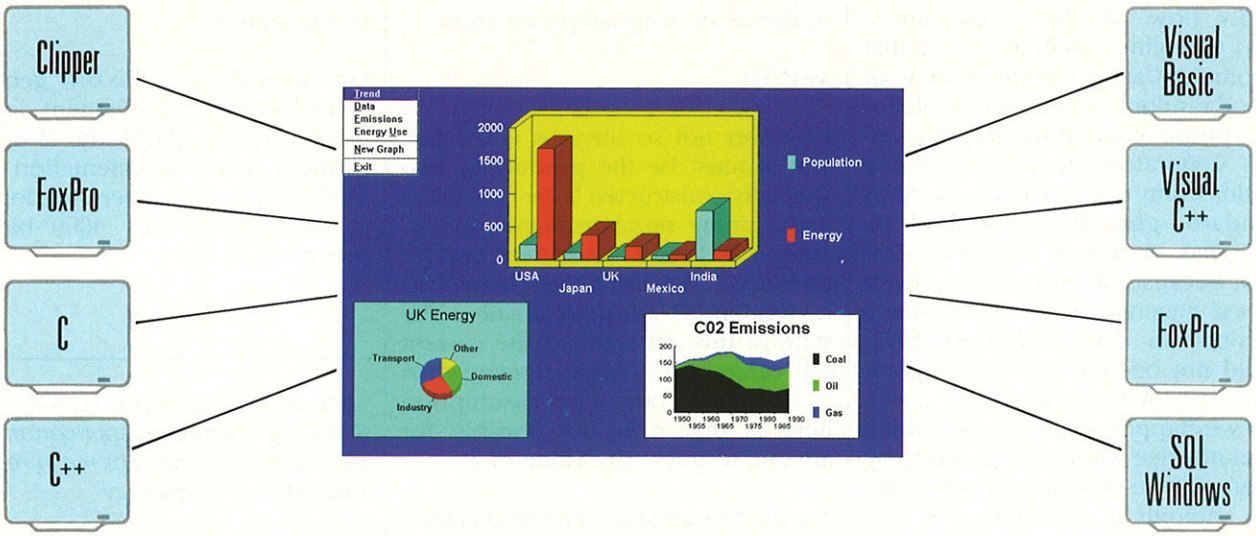
## Measuring randomness

As it stands the output from the program is a simple good/bad indicator. To see how good or how bad, uncomment the `printf()` statement in the `maurer()` routine. Initially the program tries to find an occurrence of each 8-bit value, and stores its position in a table. If a particular value does not occur within the first `Q` values, the generator is labelled as bad. Thereafter the gap between occurrences of each number is noted, and used to build up a figure of merit for the generator. If this figure deviates too much from its expected value, the generator is condemned as bad. The serious user is strongly advised to obtain Maurer's paper, as many variable parameters of this test have been fixed for the purposes of providing a simple demonstration of the method.

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

/* Maurer's Universal test */
/* for Random bits */

#define Q          10000     /* > 3000    */
#define K          1000000L /* > 100*Q   */
#define MEAN       7.1836656
#define DEVIATION 1.5*sqrt(3.238/(double)K)

int maurer(gen)
int (*gen)();
{ /* Universal Test for Randomness */
  double sum,ftu;
  int i; long n;
  static long tab[256];
  for (i=0;i<256;i++) tab[i]=(-1);
  for (n=0;n<Q;n++)    tab[(*gen)()]=n;
  /* check each byte occurred */
  /* at least once */
  for (i=0;i<256;i++) if (tab[i]<0) return
0;
  sum=0.0;
  for (n=Q;n<Q+K;n++)
  { /* scan byte sequence */
    i=(*gen)();
    sum+=log((double)n-tab[i]);
    tab[i]=n;
  }
  ftu = ((sum/(double)K)/log(2.0));
/*****************************************
  printf("ft= %lf DEV= %lf\n",ftu-MEAN,
                                DEVIATION);
*****************************************/
  if (ftu>(MEAN+DEVIATION) ||
    ftu<(MEAN-DEVIATION)) return 0;
  return 1;
}

int generator()
{
  /* Random bit generator. */
  /* Pack bits into a byte */
  int i,x;
  for (x=0,i=0;i<8;i++)
    x |= ((rand()&1)<<i);
  return x;
}

main()
{ /* test bit generator for randomness */
  if (maurer(generator))
    printf("This seems to be a GOOD random
bit generator\n");
  else
    printf("This is a BAD random bit gener-
ator\n");
}
```

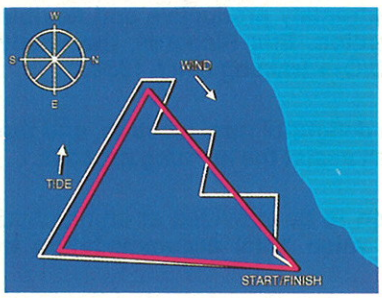
*Figure 1 - The Universal Random Bit tester*

# Programmable Graphs




**Whatever language you have chosen for DOS or Windows development, you can use one of our graphing libraries to integrate graphs and charts into your application**

dGE has been voted **Best Graphics Tool** for the last five years in a row, and last year Graphics Server won **Best DLL**.




You can create the full range of business and scientific graphs with just a few lines of 'AutoGraph' code, or control every aspect of your design with the 150 functions in the low-level API.

**'Graphics Server is the most sophisticated of these programs and would be my choice for general use'**

*review of five graphing programs in Windows Tech Journal April '93*

## NEW in dGE

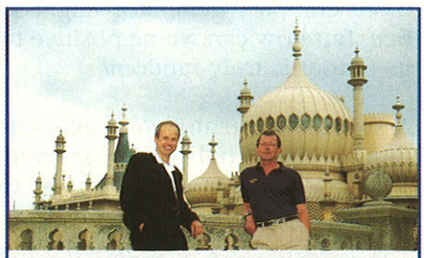
- **Super VGA support**
- **256 colours**
- **proportional fonts**
- **log/lin, bubble graphs**
- **curve fitting**
- **improved memory use**
- **support for PCX**
- **Windows bitmap support**

## NEW in GRAPHICS SERVER

- **FLL for FoxPro**
- **Chartbuilder control for VB (more features than VB Pro)**
- **new graph class for VC++ (with source code)**
- **high-level AutoGraph API (with source code)**

*PS. dGE code will run virtually unmodified under Windows with Graphics Server.*

**Program your own Executive Information System (complete with drill-down) using the 'hot' graphs in Graphics Server and Chartbuilder.**



*We founded Bits Per Second, here in Brighton, ten years ago. We started by developing graphs for a dBASE application – and we've never looked back.*

*We have become the leader in our field – so much so that Microsoft uses our software for the graph control in Visual Basic Pro.*

*If you want to do more with programmable graphs and charts (and do it royalty free), join the thousands of developers in business and science who are already using our software. Give us a call.*

*John Marsh* *Graeme Knott*

**dGE 5.0 and Graphics Server 2.20 are £245 each. Graphics Server includes the Chartbuilder graph control, available separately for £85. Our products come with a 60-day money-back guarantee.**

**CALL 0273-727119 or**
**FAX 0273-731925**

**BITS PER SECOND**



Bits Per Second, 14 Regent Hill, Brighton, BN1 3ED

## Blum, Blum, Shub...

Finally how can we implement a good gambling machine - good that is from the Casino's point of view, or a good random bit source suitable for encryption? None of the pseudo-random generators mentioned above would be any use, not even the ANSI 'standard' generator described by Kernighan & Ritchie (Second Edition), because of early cycling. Even a *good* generator such as the one I described in *.EXE* November 1992, would not be suitable - a computationally well-equipped gambler/wiretapper might guess which generator we were using. And, by observing the random sequence for long enough it would not be too difficult to determine the parameters or 'state' of the generator, and hence reveal all its future behaviour. A more complex generator might suffice. But how can we be positive that its output is truly random?

A suitable generator *does* exist - the Blum, Blum, Shub (BBS) generator. The random bits output from this generator are taken from the least significant byte of the number generated by the deceptively simple iteration:

$x=x^2\%n$

However not so obvious is the fact that $n$ must be the product of two specially constructed large (256-bit at least) prime numbers. Once a suitable $n$ is found, its factors can be discarded. (See the *Modern Crytology* by Brassard [1] for more details.) The state of this generator is the current full value of $x$, a large (greater than 512-bit) number. The assumption here is that it is not feasible for anyone to guess the value of $x$.

Are the bits generated unpredictable? Yes, because it has been proved that any successful method which could be used to predict these bits, could also be used to find the factors of $n$. It is well known that factoring huge numbers is widely regarded as impossible. So the unpredictability of the bits is anchored firmly on the bedrock of the 'hard' problem of integer factorisation. Thus we can neatly sidestep the requirement to carry out any tests for randomness on this generator.

On the downside, the BBS generator is awkward to implement in software, and is painfully slow. For example my implementation on a 33MHz 486 PC delivers random bits at the rate of only 4000 bits per second.

EXE

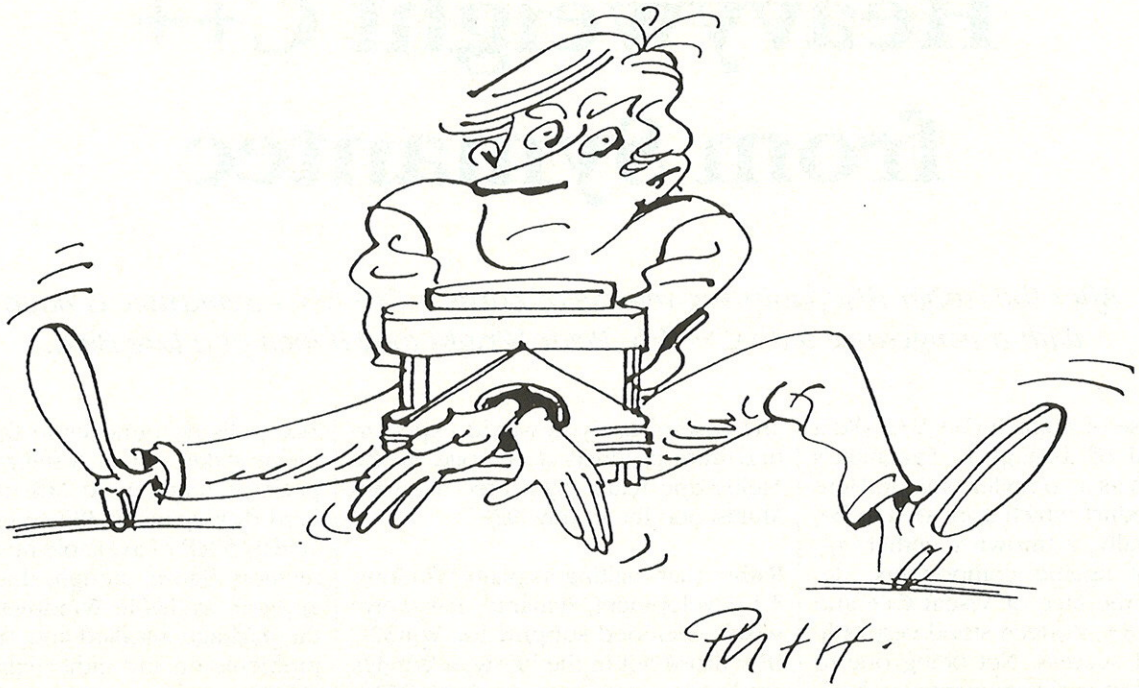
*Michael Scott is Senior Lecturer in the School of Computer Applications, Dublin City University. His main research interest is in Cryptology*

[1] BRASSARD, G. Modern Cryptology. Lecture Notes in Computer Science, Vol. 325. Springer-Verlag 1988

[2] MAURER, U. A Universal Statistical Test for Random Bit Generators Journal of Cryptology, Vol. 5, 1992, pp89-105

[3] SCOTT, M. The Ultimate Random Number Generator, .EXE Magazine, November 1992

# Hardlock E-Y-E®- tying the hands of software pirates



**The effective way to protect your software**

FAST Electronic has made life a lot harder for software pirates. Hardlock E-Y-E was designed using cryptographic principles. It took the experience and know-how of Germany's number one in software protection and the leading edge technology of a US semiconductor company to create the ultimate software protection tool.

**The technology programmers have at their fingertips**

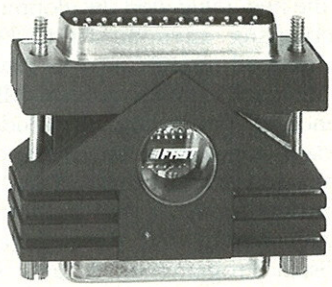
Hardlock E-Y-E is based on a custom chip and combines all the features that a programmer would expect from such a device: secure, algorithmic query routines and an optional non-volatile memory for custom configurations. With the Crypto-Programmer card from FAST you can program the algorithmic parameters and the memory within seconds. This unique card guarantees that no one else can burn Hardlock E-Y-Es with your codes. Linking Hardlock E-Y-E to your software is easy: you can either protect your .COM and .EXE files with the automatic encryption software HL-Crypt or integrate FAST's high level language routines into your source code.

**Readily acceptable to your customers**

Hardlock E-Y-E allows unlimited backup copies of the master floppy. The customer gets the device together with the software and plugs it into the parallel port between the printer and the PC. Daisy-chainability, outstanding reliability and the compact high tech design guarantee that your customers will accept Hardlock E-Y-E.

**The benefits your management will appreciate**

Hardlock E-Y-E can be programmed by the software house with the Crypto-Programmer card. This ensures optimum delivery schedules and stock flexibility. Revenues will go up as software piracy and multiple usage are prevented.



**Hardlock E-Y-E programmable, algorithmic response and memory option – all in one**

MAGNIFEYE

**Order your demo unit today. Contact Magnifeye, 235-239 Walmer Road, Walmer Studio # 6, W 11 4 EY, Telephone 071 221 8024, Fax 071 792 3449.**

**Magnifeye is a subsidiary of Fast Electronic GmbH.**

# Heavyweight C++ from Symantec

*After the rather disappointing release of Zortech C++ V3.1 Symantec is back with a vengeance with C++ V6. Peter Wright took a look at a late Beta.*

The release of Zortech C++ V3.1 did a great deal of damage to Symantec's reputation as a compiler builder. Here was a product which appeared to be, quite literally, a thrown together collection of ageing components. Released in the face of Visual C++ and Borland C++, Zortech stood very little chance of success. Not being one to admit defeat easily, Symantec is back with a completely new, redesigned product. The company's change of tack has paid off greatly; Symantec C++ V6 (SC6) is a definite heavyweight contender in the modern Windows compiler market. Microsoft and Borland would do well to panic a little. This article highlights its new features.

## First Glance

Symantec is aiming SC6 direct at Microsoft's home ground. The user interface of the new integrated development and debugging environment (IDDE) bears a marked resemblance to Visual Basic. Microsoft's much praised MFC V2.0 windows class package is included as standard (whether or not this was a wise licensing deal on the part of the big MS remains to be seen). And Blue Sky's rapid prototyping and development tool, Visual Programmer, is also included, bringing Visual Basic-style development to C and C++ programmers.

In addition Symantec has finally released a product which incorporates all the technical know-how which the company has spent the last 18 months acquiring. The full windows resource kit shipped with the compiler bears an uncanny resemblance to the Whitewater resource kit it purchased in July '92. And the three debuggers included in the package have all the functionality and ease of use you would expect to find from products developed by the Multiscope team (Symantec acquired Multiscope Inc in July '92).

Rather than halting at plain Windows 3.1 development, Symantec has (very) wisely included support for Win32s. (For those not in the know, a Win32s application can run on both Windows NT and Windows 3.1, without recompilation.) DOS programmers have not been forsaken in the sprint to the Windows altar either. A full set of command line utilities is included, as is a very comprehensive set of libraries covering all aspects of DOS development. These include libraries for memory management, TSR creation, swapping, display handling, mouse control and more.

SC6 pulls no punches in the requirements stakes either. Installation of the package requires 80 MB of precious hard disk space, a 386 or greater and at least 6 MB of available memory. The memory figure, though, does actually apply to available Windows memory; the package installed and ran with no problems on my rather sparse 4 MB 486SX.

## IDDE

As you can see from Figure 1, the IDDE is a far cry from Zortech's DOS based Workbench (ZWB). Owing a great deal more to Visual C++ and Visual Basic than I think Symantec would care to acknowledge, it represents, in my view, what a windows development environment should feel like.

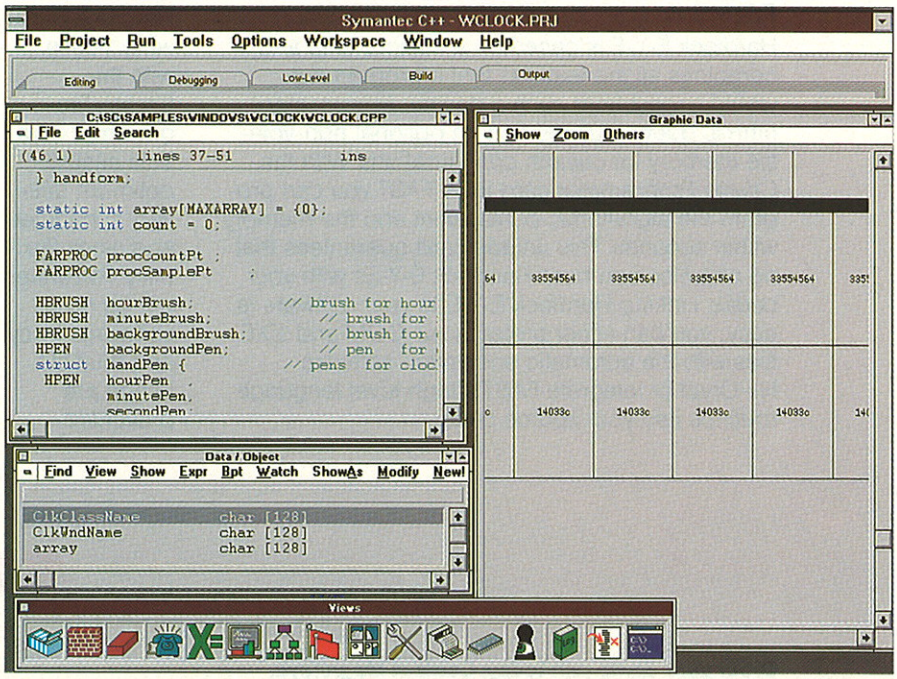
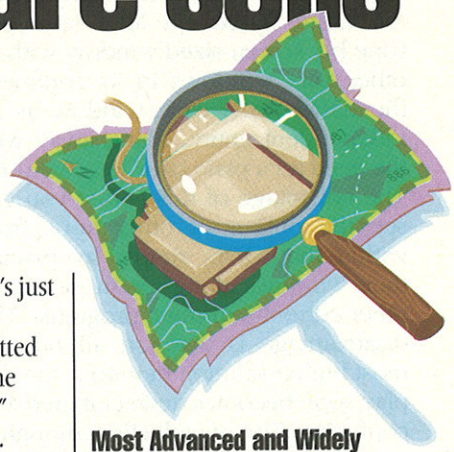



*Figure 1 - The new IDDE, compact and bijou, but highly functional*

# Why some software sells more than others.



Success. All software developers strive for it. Now, Don Gall was on top of the world. Software protection made all the difference. Especially in Europe and Asia. Sales were four times better than before. He is the founding father of Sentinel — the guru of software success.

### Struggling Software Sales

One day, trekking through the coffee fields of Java, Don ran into his old college buddy Simon Seagull. "Don, my sales are well below expectations." Simon explained his plight, "My software should sell like yours, Don!" Yet despite critical acclaim Simon's company, SimonSays Software, teetered on a financial tightrope. "What's your secret, Don?"

They spent hours analyzing potential problems. They looked at everything.

### The Key to the Problem

Finally, Don leaned back and asked the assumptive question, "What about protection – are you using Sentinel?"

Nervously, Simon sipped his coffee. His hands shaking as his eyes darted the room. "No. I didn't think I needed to."


Don's chair slid out from under him and he crashed to the floor. Amazed in disbelief, Don cried, "You What?!" Grabbing his tattered scrapbook, Don pulled out photos of his travels. "Ever been to Seoul? Prague? Anywhere? Ten bucks will buy you anything, even bootlegged copies of software."

### Don's Road to Success

Thumbing through the scrapbook, Don shared his experiences. "Back in the '80s, I was in your shoes – beaten, battered and bruised." Simon listened. "Then, after a heart breaking trip around the world, I called the Software Publishers Association (SPA)."

"I could hardly believe it. They told me developers lose billions of dollars each year. Why? Illegally copied software. In some countries there are nine pirated copies for each legal copy sold."

Simon was disgusted, "It's just not fair."

"That's why I committed myself to solving the piracy problem," explained Don.

Simon's eyes lit up. "The dongle!" he shouted. Don corrected him, "Not just any dongle — the dongle that paved the road to success for over 10,000 developers worldwide — Sentinel."

### Successful Developers Use Sentinel

Don pulled a stack of letters out of his gunny sack. "All of these people tell the same story." Don read about a successful developer from California who swears she wouldn't be in business without Sentinel. Another company says protection costs less than litigation, plus they don't have to spend time and money supporting illegal users.

Others confessed they wouldn't market products internationally without protection.

The hours flew by, story after story, Simon learned Don Gall's secret. To succeed is to protect. To protect is to secure with Sentinel.

### Most Advanced and Widely Used Dongles in the World

Backed by the world leader in software protection, Rainbow Technologies, the Sentinel Family of hardware keys is the most diverse and comprehensive selection available. For DOS, Windows, OS/2, Macintosh, LAN, UNIX and others. They're simple to install, and are the most reliable and compatible available.

Rainbow offers just-in-time delivery and the largest technical support and engineering staff in the software protection industry.

Call Don Gall today for a free copy of **"The Sentinel Guide to Securing Software."** Or better yet, ask him for a low cost Sentinel Evaluation Kit — complete with a working dongle!

*For your free guide to securing software, call your local office or distributor listed below.*

SENTINEL
*Securing the future of software*

## When you need a dongle, you need Sentinel. The only dongle Don Gall will use.

RAINBOW TECHNOLOGIES

**RAINBOW LTD./U.K.** 44 932 570066 ■ **RAINBOW MICROPHAR/FRANCE** 33 1 47 38 21 21 ■ **RAINBOW GmbH/GERMANY** 49 89 961 3051 ■ **PACIFIC RIM/USA** 714/454-2100

*Worldwide distributors:* **ARGENTINA:** Agri-Aid, S.A. 54 1 46 8364 • **AUSTRALIA:** Fagan Microprocessor Systems 61 3 699 9899 • **BENELUX:** Introcom Security Key 31 74 430 105
**BRAZIL:** Mips Sistemas Ltda. 55 21 507 1839 • **BULGARIA:** KSIMETRO 3592 79 14 78 • **CHILE:** ChileSoft Ltda. 56 2 639 8892 • **COLOMBIA:** Construdata 57 1 610 8549
**CZECHOSLAVAKIA:** ASKON International 42 2 36 44 25 • **GREECE:** Byte Computer S.A. 30 1 923 2335 • **HONG KONG:** Beta Trading C & P 852 515 0018
**HUNGARY:** Polyware Kft 36 76 481 236 • **IRAN:** Gam Electronics 982 122 2374 • **ITALY:** BFI IBEXSA SPA 39 23 31 00535 • **ITALY:** Siosistemi 39 30 24 21074
**JAPAN:** Giken Shoji Co., Ltd. 81 52 972 6544 • **JORDAN:** CDG 962 6 818360 • **KOREA:** Genesis Technologies Corporation 82 2 578 3528
**MEXICO:** Impex Computacion, S.A. de C.V. 52 66 210 291 • **NORWAY:** Perico A/S 47 2 49 1500 • **PERU:** Computadores Andinos, S.A. 51 14 42 4888 • **POLAND:** DAGMA SP.ZO.O 48 32 525 200
**PORTUGAL:** HCR 35 1 135 61865 • **SAUDI ARABIA:** ZEDAN 966 2665 1904 • **SPAIN:** MECCO 34 3 422 7700 • **SWITZERLAND:** IBV AG 41 1 741 2140 • **SWITZERLAND:** Safe Compaid S.A. 41 2 421 0385
**THAILAND:** BCS International 66 2 319 4451 • **TUNISIE:** ICS 216 1 334 417 • **TURKEY:** BIMEKS, Ltd. 901 348 3508 • **VENEZUELA:** H&R&T Osers 58 2 261 4282



CIRCLE NO. 089

The most notable similarity to Microsoft's Visual line of products is in its use of Windows. Rather than taking the more traditional MDI approach (one big screen-sized window with all other data windows in its confines), the IDDE treats each window as an independent entity, doing away with the need for a plain backdrop window. The advantage of this is that you can see the output of other apps from within the IDDE, simply by placing their windows alongside those of any other currently running programs. The disadvantage is that, on all but the most high resolution monitors, the display soon becomes quite cluttered and confusing. The overall effect though is something remarkably similar to Visual Basic.

Once the initial confusion and novelty wears off, you find yourself left with an extremely powerful environment. Sixteen different views of your project are available by dragging the appropriate icon from an icon palette box. These views show such elements of an application as the project file list, memory dump, assembler code, source, function list, data hierarchies and so on. Thanks to a little lateral thinking emanating from Symantec, on the whole, these view windows are aware of the data in other windows. For instance, click on a function name in the function window and drag it to the source window, the view in the source window jumps to the point where that function is defined. Equally, drag a class definition from the data window into the class window and a tree is displayed showing the class's position within your program's data hierarchy.

One of the most useful features of Visual C++ is its ability to store arrangements of windows in user-defined workspaces. This principle has been adopted and made easier to use in SC6. Underneath the main title bar are displayed tabs, similar in appearance to the section markers in a Filofax. Simply click on the appropriate tab and your preferred window arrangement comes into view. Five are set up as default (with the option for you to define more as you see fit), all displayed with meaningful user-defined names.

## Compile and link

As with most modern compilers, the professional edition of SC6 enables compilation from either the IDDE or the DOS command line. The compiler itself is a highly optimised, single-pass C++ compiler utilising technology borrowed from the Zortech package. As with Zortech, compilation is noticeably quick. The speed of the package, in comparison to my former favourite, Visual C++, is noticeably faster, especially when compiling MFC applications.

SC6 incorporates a number of useful features for optimising its compilation speed. As with Visual C++, programmers can elect to pre-compile headers either selectively or globally in a project. There is also the option of including each header in a project only once. If this is not checked then each time a source is encountered calling for, say, `iostream.h`, the header file is reloaded and recompiled. This reduces free memory usage during compilation and dramatically increases the length of time a program takes to compile.

The linker included in the package is SLR Systems' Optlink V4, a linker which modestly claims to be the fastest in the world. As with the compiler, it is noticeably rapid when compared against BLINK (original Zortech linker) or LINK.

As you would expect from a professional compiler, a wealth of compile options are available, including the inevitable target processor. Instead of the norm, ie implementing x86 to 386/486 then giving up, Symantec has additionally provided direct compilation to Pentium code - good move. This represents the second prudent move in the compiler by Symantec - the first being support for, and inclusion of, the Win32s SDK. These two features alone make SC6 a great deal more useful, in the long term, than either the Borland or Microsoft offerings.

In terms of code handling, the compilers deal quite happily with most of the standards put forward in AT&Ts CFront V3.0 standards. However, as with Zortech C++, Symantec have still not implemented exceptions in any acceptable form. Templates are supported though.

## Optimisation

One aspect of its compiler which Symantec is extremely proud of is its optimising technology. SC6 supports 11 different kinds of optimisation, each of which can be individually enabled or disabled (either through command line switches or via radio buttons in the IDDE). Using a closely guarded technique known as DFA (Data Flow Analysis), Symantec claims that its optimiser can achieve a speed increase of up to 30% on computation-bound code. The company also wisely acknowledges that, in some cases, this is still not as effective as carefully crafted assembler code: the optimiser looks only at the code, not the algorithm the code is trying to implement.

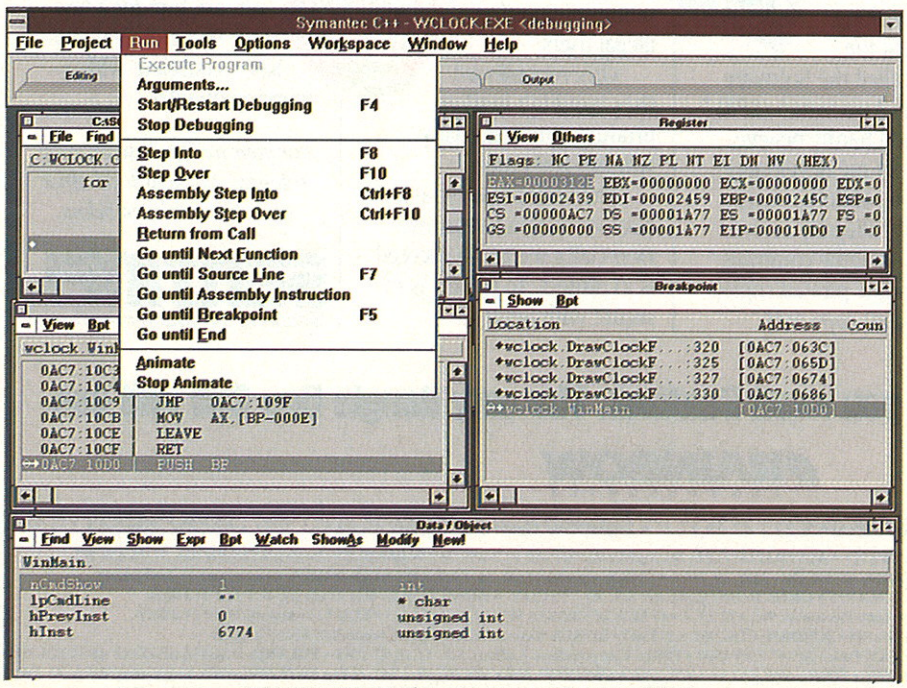



*Figure 2 - Debugging in the IDDE*

# ENTERPRISE COMPUTING

# Windows™ 93

## Expo

EXHIBITION AND CONFERENCE

**OLYMPIA, 20TH - 22ND OCTOBER 1993**

- **WINDOWS NT**
- **THE LATEST APPLICATIONS**
- **WORKGROUP SYSTEMS**
- **WINDOWS NETWORKING**
- **HARDWARE PLATFORMS**
- **DISTRIBUTED SYSTEMS**
- **MULTI-MEDIA**
- **32 BIT APPLICATIONS**
- **E-MAIL**
- **SOFTWARE DEVELOPMENT**
- **SQL DATABASES**
- **COMMUNICATIONS**

**DO YOU WANT THE LATEST ANSWERS?**

Call 0256 381045 to register for more information and your free VIP admission badge to the enterprise computing event

*sponsored by*

COMPUTERWEEKLY

The optimisations included in the package are comprehensive to say the least. Constant propagation and copy propagation are aimed at replacing variables with constants and *vice versa*, should the need arise. Also included are flags to enable elimination of common sub-expressions, dead assignments, dead code and dead variables. Loops are equally well catered for, with the ability to remove unchanging expressions from loops (loop invariant removal) and speed up array access in loops (loop induction variable replacement).

Since some optimisations may throw out previous ones Symantec have thoughtfully included the ability to loop continuously through the code, optimising until no further changes can be made. Finally, the compiler reference guide includes a 'hints and tips' section detailing further optimisations. It also deals with code that falls over after being optimised (one can imagine this might be fairly common if all options are enabled on a large, ie 25,000+ line, project).

## Debuggers

As I have said earlier, the technology for the debuggers, three of them in all, has been borrowed from Symantec's recent corporate acquisition, Multiscope. Few will deny that the Multiscope debuggers were the best available, but integration into the IDDE has curbed them a little. For instance, where in Multiscope it was easy enough to load in a .EXE for debugging, in-SC6 this is not possible unless you have the project files to hand which created the .EXE in the first place. The same is not true however of the standalone DOS debugger.

The three debuggers comprise two hard-mode debuggers, one for Windows, one for DOS, and one soft-mode debugger. The hard mode debuggers freeze all system activity during a debugging session, useful under Windows where messages may still be flying around in real time thus knocking the program to be debugged out of sync. The soft debugger is a little more traditional. Hosted in Windows, it still allows system activity in the background but being windows-hosted, it is a lot more pleasant to use than the text-based hard debuggers (see Figure 2).

Although these form the three core debuggers of the system, there is also a serial port-based remote debugger and a network aware remote debugger.

In use, all the debuggers appear to function reasonably efficiently. However, they do take a while to master. The Windows-hosted soft debugger is not too bad since its output windows are fully integrated into the IDDE and work in exactly the same way as the other windows in the IDDE. The hard debuggers try to implement a similar arrangement using character-based screens which, quite frankly, doesn't work too well. Moving windows around is at times an awkward affair with none of the polish of a Windows app. However this is a minor user interface gripe; I could find no major flaws in the debuggers themselves.

Also included are the MED and crash analyser systems previously included with the Multiscope debugger range. These two utilities could prove invaluable in a commercial environment allowing post-mortem debugging on crashed programs. It involves a little extra linking at the compilation stage but the benefits during testing and debugging could prove limitless.

## Visual Programmer

Microsoft supplies App and Class wizards with Visual C++, and Borland includes a copy of Protogen with BC++. Symantec, too, ships an application generator with its C++ compiler. This time it's Visual Programmer (VP) from Blue Sky.

VP actually works on two levels. On the first, it is a resource kit, enabling the developer to put together menu structures, dialog boxes and so on. On the second level it allows the developer to link C or C++ code direct to the user interface components in a similar fashion to the Microsoft Class Wizard or Visual Basic. The difference between these is that, in many cases, functionality can be added to a control element without the need to type anything. For instance, the ability to run a third party program is built into the system. If you needed to run such a program in response to the user clicking a button on the screen, then the developer only has to provide the path and file name of the program to run; the rest of the work is done automatically.

A novel feature of the package is VP's testing function. It is possible to design a complete user interface and then run it, without compiling, to see how it works and fits together. If something doesn't look right then it can be changed while the app is running, enabling you to see instantly how the change affects the program.

As with everything else in the package, VP is tightly integrated with other components of SC6. Select compile from the VP menu and SC6 is loaded up and a compilation run. Elect to enter code to link with the user interface and the

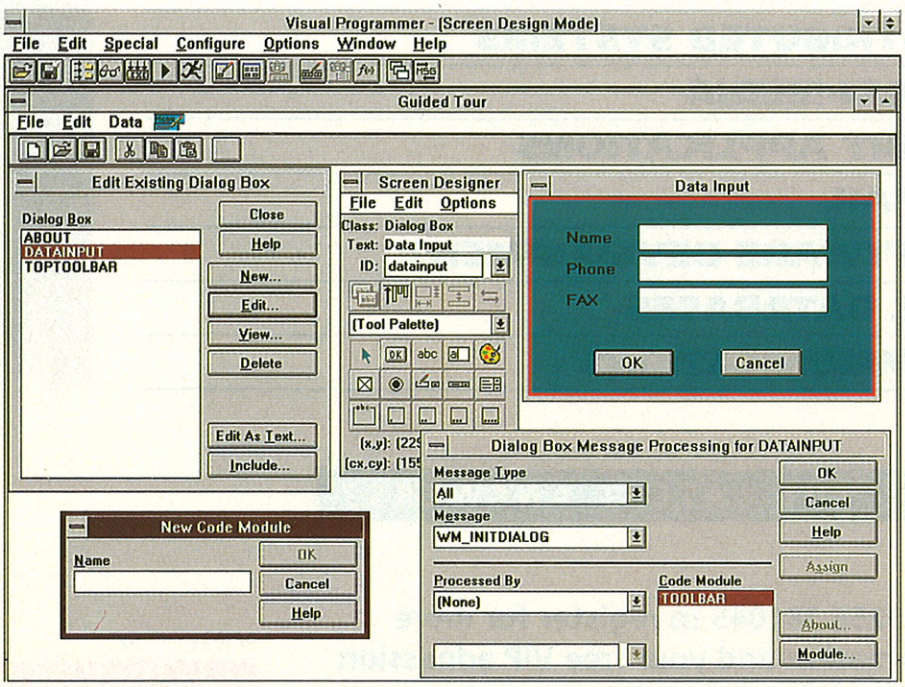
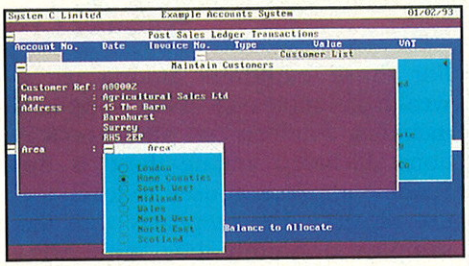
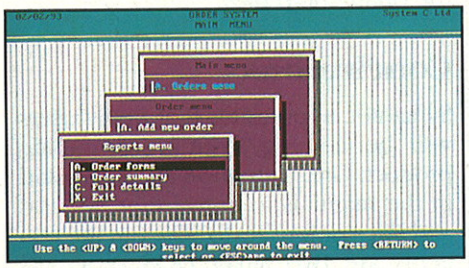



*Figure 3 - Blue Wave's Visual Programmer system is included as standard*

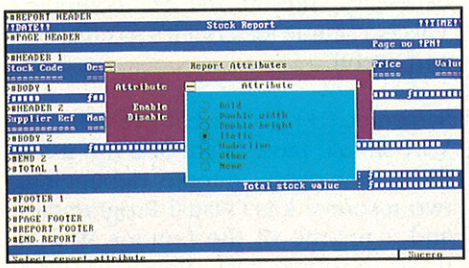



Sycero supports panels, movable windows, VGA colours, scrolling screens, tick boxes, radio buttons and much more

# *I'm so* BRILLIANT




Use the definable program templates to produce different program types including menus, file maintenance, reports and enquiry programs




The report generator automates layout design, nested loops, print attributes, ten levels of sub-total, range checking, wild card matching and up to 99 formats per report

*that I don't need an applications generator* - A typical cry from thousands of developers who still program manually.

Remind you of anyone?

In fact, Sycero, an applications generator which develops Clipper database programs, has been helping over 12,000 developers of all levels of expertise program more effectively since 1983. And in a recent survey, the vast majority said that programming with Sycero was between 2 and 5 times faster than manual coding.

Sycero 3·0, an all-new release, is even faster and easier to use. Its data dictionary organises your data files and indexes. The new screen painter lets you design a modern, elegant interface that end users will love. No database application is complete without reports, and Sycero's powerful combination of automatic reporting features makes even the most complex ones simple to produce.

## SPEEDS UP CA-CLIPPER PROGRAMMING

The key to Sycero's success is its flexibility. The ability to add Clipper or Sycero code at dozens of trigger points within the development environment means that Sycero-generated applications can be as flexible and powerful as those developed by coding manually in Clipper.

So, even brilliant programmers can use Sycero to reduce development times, improve the user interface of their applications, produce complete program documentation and automate networking. Come and see for yourself at one of our regular free seminars. In just a couple of hours we will show you how you can be brilliant in half the time!

> *"Our software operates the accounts of over 4,000 farmers, so it has to be powerful, flexible and simple to use. We wrote the entire suite of 500+ programs with Sycero in just nine months. It would have taken years coding it by hand"*
>
> **Arnold Jacques** Cerco

Please send me details of Sycero 3·0 and your next free seminar

**NAME**

**COMPANY**

**ADDRESS**

**POSTCODE**

**TELEPHONE**

**FAX**

I am/am not a CA-Clipper user

**Send to:**

**System C Limited**
60-61 High Street
Maidstone
Kent
ME14 1SR

Tel: 0622 691616
Fax: 0622 691241

SYCERO 3·0

➤ CIRCLE NO. 091

user can choose to use either the built-in editor, or call up the IDDE. A screen-shot of Visual Programmer is given in Figure 3.

## Docs & Help

Since the version reviewed here was only a late Beta, the documentation supplied was far from complete. However, that which was supplied was very extensive. Both the *Compiler, Debugger* and *Visual Programmer* guides all made use of readable, easy to follow tutorials which expose the developer to most facets of all these systems in a very short space of time.

Documentation on the compiler itself was of an equally high standard. Not only was the use of both IDDE and DOS compilers covered, but great lengths were also taken to explain how these work, something normally shrouded in some mystery. Full documentation was also supplied covering the ins and outs of SLR's Optlink linker system.

On the whole the documentation was more than adequate and fairly easy going for a package of SC6's complexity. I am reliably informed that there will be even more books included in the final edition, covering both Windows APIs (3.1 and 32s). Hopefully Symantec will have the grace to include some documentation on the DOS libraries too, at the moment these are only covered in any detail in the Help system.

The help files supplied are equally extensive in their coverage of the operational aspects of the package. In addition to the obligatory MS pen, multimedia and Win 3.1 API help files, extensive online help is also supplied for Win32s and the Symantec DOS libraries. The latter covers all aspects of using these libraries, essential since no paper coverage of them was available for the review. One area that was mysteriously lacking was help for the IDDE. General overviews of the main tool palettes were provided, but that was about all. Hopefully this will be rectified in the final version.

***Symantec C++ V6.0 represents the culmination of 18 months worth of corporate acquisitions: Zortech, Whitewater, Multiscope...***

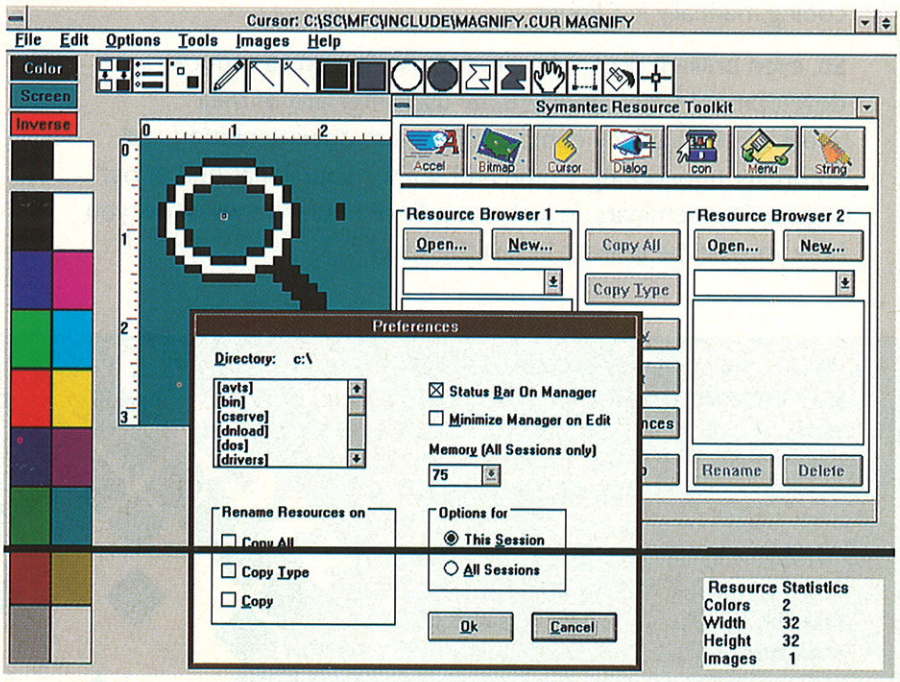



*Figure 4 - The Whitewater, sorry, Symantec Resource kit in action*

For those explorers among us, a wealth of sample code is also supplied, including fourteen full MFC applications. Included among these is the Scribble application used to teach the use of MFC. The emphasis with the samples definitely lies with MFC though. Only six straight Windows examples are included; and only one DOS program. There are quite a few further examples, however, built into the online help files.

## Conclusions

Symantec appreciates that many users will be coming from a competing product. Great attention has been paid in the manuals to converting Borland and Microsoft users to SC6. But the real gems are two small DOS programs called BCC and CL. These 'fake compilers' take a compile string for either Borland or Microsoft C, convert the syntax to that required by SC6, and, if necessary, fire-off the SC compiler. This is a simple idea but it's so effective and useful.

Symantec C++ V6 is what Zortech C++ V3.1 should have been. SC6 has it all: three debuggers, remote debugging, two resource kits (Visual Programmer and a rewrite of the famous Whitewater resource kit), Windows- and DOS-hosted compilers, the IDDE linker and DOS Optlink linker, MFC V2.0...

Symantec C++ deserves pride of place in any serious C/C++ programmer's catalogue. It has all the hallmarks of a package that looks set to stay. It's just my opinion, but I believe Symantec C++ V6.0 is the most powerful C++ development system available on the PC today.

EXE

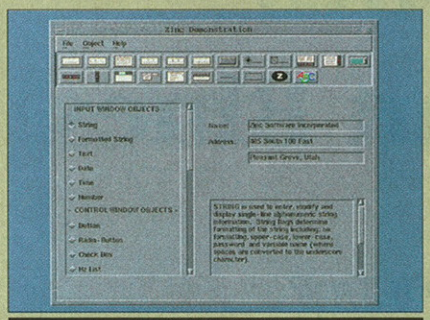
---

*Symantec C++ V6.0 was supplied by Symantec (0628-592222). UK prices were unavailable at the time of writing. The final version should now be shipping.*

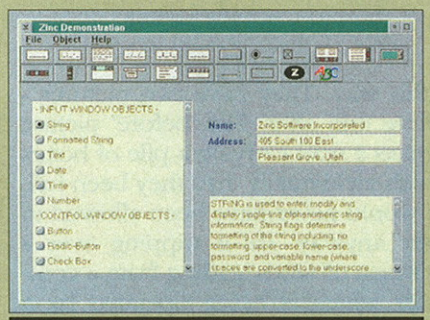
*Peter Wright is a professional programmer and software consultant, with slight journalistic tendencies. He can be contacted on CompuServe using ID 100116,357, or via his company, Genesis Software Consultancy, on 0323 507664.*

OBJECT-ORIENTED · MULTIPLATFORM · SINGLE-SOURCE · C++

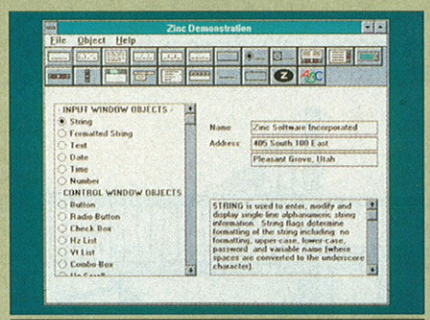
THINK ZINC



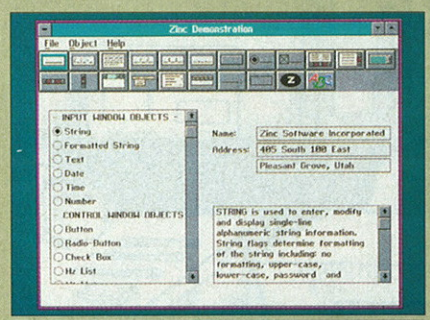
UNIX MOTIF



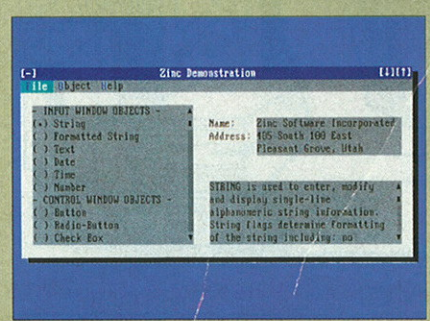
OS / 2 2.0



MS WINDOWS & WINDOWS NT



DOS GRAPHICS



DOS TEXT

ZINC™ APPLICATION FRAMEWORK™ 3.5

# Multiplatform Flexibility From One Application Framework.

**FLEXIBILITY** *is an essential component in software design. Your development tools should allow you to easily incorporate new features and technologies into your applications without rewriting all of your code. That's a tall order if your development tools are designed like a straight-jacket. Zinc Application Framework 3.5 and Zinc Designer™ give you flexible and extendible support for Microsoft Windows, Windows NT, OS/2 2.0, UNIX Motif, DOS Graphics and DOS Text. And Zinc does it with ONE set of source code. Zinc's multiplatform, object-oriented architecture won't confine your application development options today... or tomorrow.*

**FOR A FREE ZINC DEMO KIT, CALL US TOLL FREE TODAY AT**

**1.800.638.8665. IN EUROPE CALL +44 (0)81 855 9918**

Zinc

ZINC SOFTWARE INCORPORATED, 405 SOUTH 100 EAST, 2ND FLOOR, PLEASANT GROVE, UTAH 84062, TEL 801.785.8900, FAX 801.785.8996, BBS 801.785.8997.

EUROPE: ZINC SOFTWARE (UK) LIMITED 58-60 BERESFORD STREET, LONDON SE18 6BG, TEL +44 (0) 81 855 9918, FAX +44 (0) 81 316 7778, BBS +44 (0) 81 317 2310



CIRCLE NO. 092

# Computer supplies - The RS232 Breakout Blues

*The catalogue season is with us - indeed when is it not? Jules has been browsing a few which have been sent to him.*

They say everyone remembers their first time. I remember mine. I was still at school, and was gaining my most formative experiences on an ageing PDP-8, twice my size. Bear in mind that, all those years ago, the idea was bizarre that a computer was something comprehensible - I suppose I thought that computers simply materialised, fully-formed. A gift from the gods. One day, I walked into the machine room, and there it was, just lying on the table for anyone to find. I had discovered my first computer supplies catalogue.

This was something of an eye-opener for me. As I turned the pages, product after product appeared. I had no idea what most of them were, each one illustrated in expensive colour printing and announced with huge, screaming headlines. Was I reading a book about computers, or was I reading *The Sun*?

The same company, and a few others, are sending me their catalogues now - on average, three from each one. (I rang and asked them to stop sending me so many; instead they sent me more!) It still has products ranging from the mundane to the totally useless. It has, for example, 10 pages of *diskettes*. Each line is illustrated in full colour with a photograph comprising a box of disks (in full livery), and a few individual disks carelessly strewn in the foreground, presumably so you will know what a floppy disk is supposed to look like, and perhaps to underline the point that the contents of all these boxes are indistinguishable in any case. Its harder to buy disks from these people than it is to buy washing powder! But, this gratuitous waste of paper and ink is as nothing when compared to some of the other entries.

This same catalogue sells a cable tidy, which sits on the floor and prevents people from tripping up. It's a reasonably useful product - if your office design is so bad that you need one, I guess you'll know that you need one. But the catalogue company doesn't see it this way. They have two pictures - before and after. The 'before' picture depicts a man carrying a pile of boxes which incidentally, had they been full, he would never have been allowed to carry anyway. He is tripping over a rat's nest of wires leading from a computer which is being operated by a very shocked-looking secretary. Why she is shocked is beyond me: he must have tripped over those wires a dozen times that day. Neither of them appears

to be capable of learning anything. Nevertheless, the 'after' picture shows the same man, with the same precarious pile of boxes, completely *failing* to trip over the wires. Ah, so that's why we use cable tidies! I'm *so very glad* it was explained to me.

Before and after pictures are a very common ploy. They have before-and-afters for anti-glare screens (obvious), antistatic carpets (even more obvious), acoustic printer hoods (slightly less obvious), and even burglar deterrents. What these pictures all show, apart from the product in use, is the comic-book emotions of the protagonists.

However, the programmable, intelligent magic box, headlined 'Solve your RS232 breakout blues', defied even the photographer's ingenuity. Presumably, the before picture would depict a harassed engineer, surrounded by miles of unmarked wire, with a concentrated frown on his face; the after picture would depict a harassed engineer, surrounded by miles of unmarked wire, with a concentrated frown on his face...

The point of this piece is not merely to pour scorn on these idiotic catalogues. You're a programmer, right? I bet you think you're pretty darn smart - far too smart to fall for this kind of clumsy, amateurish manipulation. I'm equally sure that everyone who receives these catalogues thinks the same thing - that they are unaffected by it. I'm even sure that most people find this kind of advertising patronising in the extreme. Why, then, do the companies spend so much time, money, and paper on alienating their customers? I think the answer lies in the grimaces on the faces of the models.

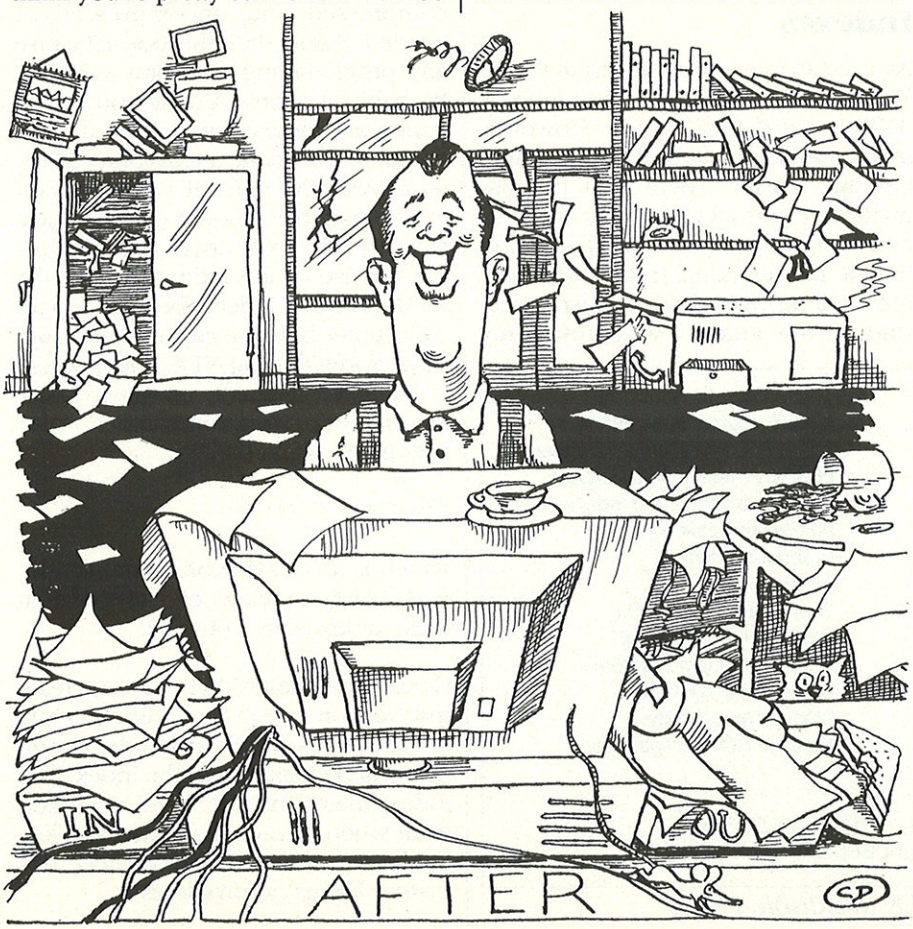
***It's harder to buy disks from these people than it is to buy washing powder!***

Most computer installations comprise lots of boxes which do all sorts of arcane jobs, the mains wires for which represent a significant cabling problem on their own. Most of these installations are just beyond the limit of comprehensibility of their minders - systems from the smallest to the largest tend to expand until any further tampering upsets them as much as it improves them. When a problem arises, a solution has to be found quickly, easily, and without removing any covers.

This is the market the catalogues hope to address. When you need something in a hurry, when you have valuable records at stake, and you don't trust your machinery, it is easier to find something colourful and glossy. At this time, when you're thinking about your problem, and not concentrating on the pictures at all, their message sinks in, and you identify with the models in the photograph. You see the secretary being polite to callers, the manager smiling for the first time in his life, and you feel better - before you've even picked up the phone. In the next emergency, you will reach for the catalogue again. It doesn't matter how smart you are, if you're sufficiently distracted you will see them only out of the corner of your eye. When you're calm again, you can feel as alienated as you wish.

It doesn't take much planning to avoid having to use these dreadful catalogues, but I can't help wondering why other emergency services don't use the same techniques. Dr Solomon or Norton Utilities could easily be sold this way, possibly generating even more revenue by making sales to people who are barely able to use them. Companies could explain the quality of their helplines using the same technique. The really disturbing possibility, though, is that of services invented specifically to be sold this way. The *late-night sandwich and coffee delivery* service, the *clean up after the spontaneous office party* service, or even the *counselling for the programmer who still can't find that dangling pointer* service. One day everything will be sold this way. What a horrible thought!

EXE




---

*Jules' latest enterprise, the* **find the service you need** *service, can be contacted on 0707 644185, or on CIX as jules. For a fee, he will arrange catalogues to be sent to you, and will even recommend suitable or interesting pages by telephone.*

# RDD Roundup

*Guy Smith provides an overview of RDDs. What's available at present, how different RDDs work and how to write RDD-independent code.*

CA-Clipper V5.01 introduced the concept of the RDD (Replaceable Database Driver) for enhancing the database capabilities built into Clipper. At the time it didn't mean much, as the only full RDD available was DBFNTX. Others were available, but these were transfer RDDs (eg SDF, DELIMITED) which meant they could only import and export between different file formats. Beginning last year and now in full flow, RDDs have come of age. Figure 1 shows a list of RDDs available at the time of writing. And there's more coming. Of course this list doesn't include the vast number of API libraries which have not been made into RDDs.

## Usage

There are several ways to use an RDD other than DBFNTX (the default for CA-Clipper applications). The first is to link in a special .OBJ file supplied by the third party vendor which forces DBFNTX out and replaces it with the new RDD. The second is to use the `dbSetDriver()` or `RDDSetDefault()` function in the application to load the RDD explicitly at run-time:

```
* make DBFCDX the default
dbSetDriver('DBFCDX')
```

There is no difference in purpose or syntax between `dbSetDriver()` and `RDDSetDefault()`. The former is present only for compatibility with Clipper V5.0; the latter is only available in V5.2.

Another alternative is to specify the name of the RDD whenever file-based commands or functions are used with the `VIA` clause:

```
USE MYFILE VIA "DBFCDX"
COPY TO TEMP VIA "DBPX"
```

The last method is the least attractive as it creates dependencies in your code: I would recommend you avoid it since you want to be writing as much reusable code as possible.

## Indexes

As you can guess from the list in Figure 1, most RDDs, at present, use good ol' .DBFs as the data file format. However, many popular RDDs replace the indexing mechanism. There is a popular myth that you can run your applications with a new RDD simply by slotting it in, relinking (and sometimes recompiling first). This just isn't true. Later in this article I will present my evidence. For now, I'll look at some of the properties of RDDs. The most common feature of new RDDs can be summarised in the three'C's:

- Compound
- Compact
- Conditional

Compound means that, instead of putting each index order in a separate index file (eg .NTX), several index orders are placed in a single file (eg .CDX).

Compact means that indexes are stored in compact form. This is seamless. So far as the application programmer is concerned all index data is compressed and uncompressed as needed. Surely then this doesn't cause any programming problems you say? Probably it won't. But if you create temporary indexes on a RAM drive it might. If you create a temporary index on a RAM drive you will certainly want to compare the amount of disk space free on the RAM drive with the expected size of the index to see if the RAM drive can, indeed, be used. Figure 2 illustrates how the calculation would be performed using NTX. The problem is that there is virtually no way to know the size of a compound index. Your best guess is:

```
nReccount * nKeyLen
```

which is a considerable overestimate, particularly in view of the fact that these indexes are compact.

Conditional means that you can create indexes on a `FOR` condition. Only those records which match the condition will be included in the index. The index maintains the `FOR` condition even when records are added and key fields are changed. Fortunately this feature brings no problems.

| | | | |
|---|---|---|---|
| DBFNTX | CA-Clipper 5.01, 5.2 | - | The faithful standard |
| DBFNDX | CA-Clipper 5.01 (US), 5.2 | - | dBASE compatible indexes |
| DBFCDX | CA-Clipper 5.2 | - | FoxPro compatible memo/indexes |
| DBFMDX | CA-Clipper 5.2 | - | dBASE IV compatible indexes |
| DBPX | CA-Clipper 5.2 | - | Paradox 3.5 data and indexes |
| DBFSIX | The SIx Driver | - | FoxPro compatibility |
| DBFCMX | Comix | - | Proprietary indexes |
| DBFNSX | Hiper-SIx | - | Proprietary indexes |
| DBFNTXAX | Advantage NTX | - | Client/Server DBFNTX |
| DBFCDXAX | Advantage CDX | - | Client/Server DBFCDX |
| FLEXFILE | FlexFile RDD | - | Proprietary memo system |
| RDDKIT | The RDDKit | - | Collection of 6 RDDs |
| RASQLB | RaSQL/B | - | Btrieve compatibility |
| RASQLX | RaSQL/X | - | Netware SQL compatibility |
| *Special cases:* | | | |
| MachSIx | RushMore-like query optimiser for DBFSIX and DBFNSX | | |
| ClipMore | RushMore-like query optimiser for DBFCMX | | |

*Figure 1 - List of RDDs available today*

Pundits often claim that it is easy to use a different RDD by replacing all:

```
if file('MYINDEX.NTX')
```

with:

```
if file('MYINDEX'+indexExt())
```

and relink the application.

This is a short-sighted translation. There's a lot of code out in the real world which works as follows:

```
for n:=1 to nNumberOfIndexes
  erase ('SALES'+str(n,1)+;
         indexExt())
next
```

Clearly, if you change from a single order 'bag' with one index per file (eg .NTX) to a multiple order 'bag' which stores several indexes per file (eg .CDX), code like this will not work. This is particularly relevant for all those routines which perform 'self- healing' of indexes (ie when the application starts up, it checks to see whether indexes exist and, if not, it rebuilds them). It is also relevant if you use a data dictionary of any kind.

## Tags

Now the introduction of compound index files means a few changes to your code. In single order bags before V5.2, you would change the controlling index using:

```
SET ORDER TO 2 or:
dbSetOrder(2)
```

You can still do this with compound indexes, but your application will not behave consistently. The problem is that index orders (or 'tags') move around inside the index order bag. When you create all the indexes, the first order may well be first in the index bag, the second order second, and so on, but it won't stay that way for long. Every time you reindex the tag, the order will be put at the end of the list. So `SET ORDER TO 2` may get a different order from one run of the application to the next. For this reason most RDDs now allow the setting of a controlling index by a name (a 'tag' name) instead of an order:

```
SET ORDER TO TAG STOCKID
```

This also adds to the readability of the code and eliminates the problem of deleting an index in a list and having to renumber all higher indexes in the source code. However, changing to tags also means that it is a one way trip for CA-Clipper V5.01 programmers: DBFNTX in V5.01 has no concept of tags.

The lack of permanence of the relative position of index orders to each other has a knock-on effect: the behaviour of DBFNTX to issue a `SET ORDER TO 1` automatically when indexes are opened has no meaning. As a result, the default order when multiple order bags such as CDX are opened, is `0` (ie natural order). This may require adjustment of existing code to cope with such behaviour.

Due to the necessity for a tag name (instead of an index order number), all RDDs after V5.01 support tag names regardless of whether the index type allows single or multiple order bags. This improves the portability of code across V5.2 applications. However, not all RDDs which support the use of tags have tag permanence. For example, whereas you can use tag names with DBFNDX in V5.2, the tag name itself cannot be stored in the .NDX index. The index can be created with a tag name when using DBFNDX and it will respond to this tag name, but as soon as the index is closed, the tag name is forgotten. On subsequent use of the index the tag name is defaulted to the file name of the index.

```
function NtxSize(nReccount, nKeyLen, nMode)
* Purpose :- to return the maximum or minimum index size for an NTX file
*
local nMinKeys:=int(((1020/(nKeyLen+10))-1)/2)
local nBytes
nMode:=iif(valtype(nMode)=='N',nMode,1)
if nMode=2
   * calculate minimum size
   nBytes:=1024*(2+int(nReccount/(2*nMinKeys)))
else
   * calculate maximum size
   nBytes:=1024*(2+int(nReccount/nMinKeys))
endif
return(nBytes)
```

*Figure 2 - Determining the size of an NTX file*

Most RDDs which support compound indexes support multiple index types as well. DBFCDX supports both CDX (multiple order bags) and IDX (single order bags). However this creates an ambiguity in the `indexExt()` function. If two index types are supported, which should a call to `indexExt()` return? At present it returns the single order bag type (eg NTX, NDX, IDX, PCB). If you are using multiple order bags (which is, after all, one of the reasons for using a different RDD) CA-Clipper cannot help you with writing reusable code which deals with index file names.

Where multiple index types are available, the normal practice is to use the multiple order bags for permanent indexes, and the single order bags for temporary indexes. The reason for this is that a temporary index will be deleted after it is used. Space occupied in multiple order bags is not reused when a tag is deleted, so creating temporary indexes in a multiple order bag creates index 'bloat'(think of this as equivalent to 'memo bloat' for indexes). However, it is not strictly necessary to use the secondary index type for temporary indexes as many, but not all, RDDs allow more than one multiple order bag per data file. So the temporary indexes can go in the second multiple order bag. Unfortunately you have to know which RDDs support multiple, multiple order bags and which don't. Furthermore you can't simply decide to create all temporary indexes as single order bags. For instance, Hiper-SIx only supports multiple order bags.

## Production Indexes

A production index is an index which is automatically opened when the data file is opened. The index has the same name as the data file so:

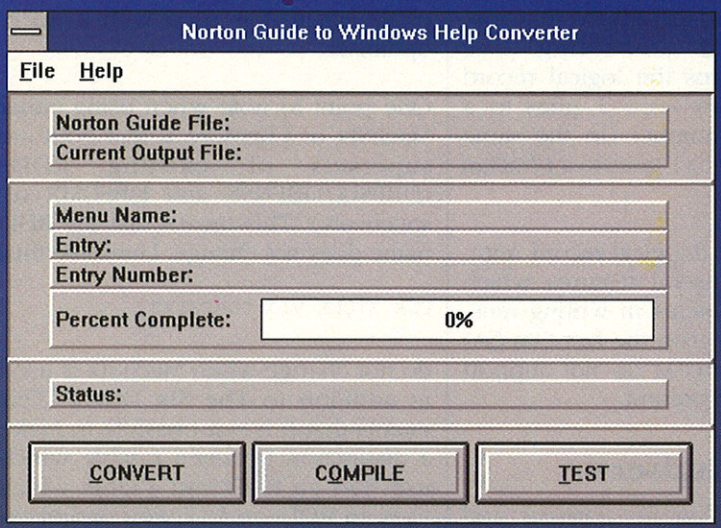
```
USE MYFILE VIA "DBFCDX"
```

opens the data file and indexes, if MYFILE.CDX exists. To date, only multiple order bags can be production indexes. This is useful as it increases the amount of reusable code. However multiple order bags make the need for production indexes less important. The following code is reusable regardless of the number of indexes associated with the data file:

```
USE MYFILE INDEX MYFILE1
```

# NG-2-HLP

## Norton Guide Users - Your prayers have been answered!




NG-2-HLP is a great new utility that enables Norton Guide users to migrate their DOS based Help Systems to Windows!

**Its as easy as 1,2,3!**

Just open your existing Norton Guide File, press the Convert Button and then the Compile Button. It's that simple!

NG-2-HLP will start working as soon as its installed - there are no complex commands or confusing configuration options to learn. If you can use Windows you can use NG-2-HLP!

**£69.95**

For Information ➤ CIRCLE NO. 095

ExpoTech
CORPORATION

*The Leaders in Development Tool Distribution*

ExpoTech House, Western Road, Hove, Sussex. BN3 1AF
Sales: +44-(0)273-749222 Fax: +44-(0)273-746446

(assuming that MYFILE1 is a multiple order bag).

It is also interesting to note that, whereas all RDDs which support multiple order bags support production indexes, not all are capable of supporting indexes which are not production indexes (eg Comix).

## Physical Existence

Some of the problems of the different types of physical existences have already been discussed. However, there is more fun to be had from this subject. I have already mentioned that if you use the `file()` function to test for existence, then the code will need to be modified if you switch to multiple order bags. What I have not mentioned is that, with single order bags, you can use the `file()` function to test for existence any time. But to test for the existence of a tag, you **must** have the order bag open first (most RDDs have a function to return a list of tags open in the current work area). This difference means that there are two levels of checking for physical existence. First, you need to test to see if the bag exists; then you need to open the bag and test for the tag.

The destruction of indexes also causes problems. With a single order bag you issue `fErase(<indexFile>)`. You will normally want to close the index first before erasing it; otherwise CA-Clipper might decide to flush the index buffers to the screen: the user will not be impressed. Conversely, to destroy a tag, the index bag **must** be open first. Consequently, it is difficult to write code which is portable between the two types of indexes.

RDDs exist for both BTrieve and Netware SQL. In BTrieve, index orders are often maintained inside the data file itself; in SQL engines, all tables and all indexes are maintained in a single file. At present RDDs provide no help in writing code which can hide these differences.

Finally there is no `fileExt()` function equivalent to CA-Clipper's `indexExt()` function for indexes. As Paradox and other formats do not use the .DBF file extension, this can lead to further problems.

## Scoping/Logical Records

Many third party RDDs (but not the RDDs in V5.2) allow tables to be scoped. They also support the concept of logical records. To set a scope use:

```
SET SCOPETOP    TO <top>
SET SCOPEBOTTOM TO <bottom>
```

For example:

```
SET SCOPETOP    TO "LONDON"
SET SCOPEBOTTOM TO "NOTTING-
HAM"
```

From this point on the table appears as if only those records which are in the scope are in the table. So `GO TOP` goes to the first record at or after 'LONDON'. This very useful feature cuts out the need to write specialised TBrowse `skipBlocks`.

Logical record numbers are very useful for scroll bars. You often need to be able to determine the current record's position relative to the top of the index order. `RecNo()` does not help here as it returns the physical record number (or indeed record identity). A function, usually called something like `KeyNo()`, returns the logical record number and `KeyGoto()` goes to a logical record number (in the same way that `dbGoto()` goes to a *physical* record number).

Both scoping and logical record numbers are very useful features which present no problems in writing reusable code other than the fact that CA-Clipper's own RDDs do not support such features at present.

## Query Optimisers

MachSix (for The SIx Driver) and Hiper-SIx and ClipMore (for Comix) both provide query optimisation similar to FoxPro's RushMore. Query optimisation is achieved by recompiling code with a special header file and relinking with additional libraries. No physical change to the source code is required in terms of syntax (other than the recompiling). A small change may be required to the methodology of multi-user applications. Both query optimisers take a snap shot of index data and perform optimisation based upon this information. If the index data changes (ie records are added by other users or index fields updated) then the `SET FILTER`, unlike normal CA-Clipper `SET FILTER`, will not reflect the changes. For this purpose both products include a way of refreshing the filter. Along with their optimisation ability, these RDDs also provide a host of other less useful (irrelevant?) commands such as `AVERAGE`, `COUNT`, `LIST` and `SUM`.

Query optimisation occurs on `SET FILTER` and `LOCATE`. The effects of either MachSIx or ClipMore are considerable. The only caveats in the usage of either product are that you should check the documentation to see exactly which expressions are optimisable and which are not. Clearly a basic requirement is that the query expression contains at least one field for which there is an index.

The result of all of this is that the `SET FILTER` command is now a credible part of the language. The contempt and scorn which the command has attracted for years is no longer appropriate when it is used with a query optimiser.

One point to note when using either MachSIx or ClipMore is that both are supersets of existing RDDs (DBFSIX/DBFNSX and DBFCMX respectively). This means that the RDD name does not change. Lines like this:

```
USE STOCK VIA "DBFSIX"
```

do not change when MachSIx is used in addition to The SIx Driver. This means that it is not possible to make a distinction between a table which uses just an RDD and a table which uses an RDD *and* a query optimiser. This is relevant for code which stores the RDD name in a data file (for data driven applications) and for code which needs to be able to determine the behaviour of a table at run-time based upon the RDD being used.

## Additive RDDs

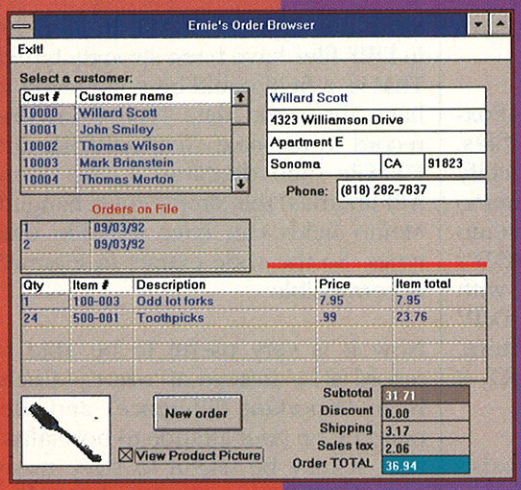
Several RDDs fall into a unique category - that of the 'additive' RDD. These RDDs are not 'host' RDDs. Instead they add functionality to an existing RDD. A few examples are give in Figure 3. Such RDDs require a 'host' RDD to provide the default behaviour. They add or modify specific behaviours. The existing RDD

| | | |
|---|---|---|
| FlexFile RDD | - | Replacement Memo System |
| The RDDKit | - | Collection of 6 additive RDDs |
| Hiper-SIx | - | High Performance Index Replacement |

*Figure 3 - Examples of additive RDDs*

# Agility

## The Database Manager for VB/VC




Create databased applications in minutes using Agility's custom controls, all without writing a single line of code!. All of Agility's controls are hot-linked to data files and are updated automatically. With Agility you can access multiple file formats: dBase, delimited text and native Agility files. Create databases in memory for look-ups, and for sharing between applications. Agility also provides an easy to use QBE interface with full query optimisation.

*AgiButton:* With One click, you can process queries, add/update records or navigate files,

*AgiText:* Build Instant forms by hot-linking text boxes right to the database. Includes edit masks for complete data input control.

*AgiGrid:* An instant, full featured browser. Editable cells, movable and resizeable columns, scroll bars, user programmable data validation, interactive design-time layout. You can full program control over grid contents/operations.

*AgiText:* Build Instant forms by hot-linking text boxes right to the database. Includes edit masks for complete data input control.

**£189**

+ Carriage + VAT

## ALSO AVAILABLE - 'TRUEGRID'

The New data-aware grid control from Apex!

**£69.95**

+ Carriage + VAT

## ExpoTech

CORPORATION

***The Leaders in Development Tool Distribution***

ExpoTech House, Western Road, Hove, Sussex. BN3 1AF
Sales: +44-(0)273-749222 Fax: +44-(0)273-746446

CIRCLE NO. 096

system in CA-Clipper V5.01 and V5.2 does not provide a standard for specifying an RDD inheritance chain, so each product differs in the mechanism it employs to add functionality.

The inheritance problem is simplest for Hiper-SIx. Hiper-SIx has a hard-coded algorithm for deciding which host RDD to use. At run-time it looks to see if DBFSIX (The SIx Driver) is loaded and, if it is, uses it as the host RDD. If not, it looks for DBFCDX (CA-Clipper V5.2). If DBFCDX is not loaded it looks for good ol' DBFNTX. This decision as to which RDD Hiper-SIx inherits from cannot be changed. In the program you would write:

```
USE STOCK VIA "DBFNSX"
```

The FlexFile RDD is a little more flexible. Note that this discussion is concerned with the FlexFile RDD and **not** the FlexFile API. With FlexFile you can specify which RDD FlexFile should inherit from using FlexFile's `V_SuperRDD()` function:

```
V_SuperRDD("DBFNTX")
USE STOCK VIA "FLEXFILE"
```

The most flexible of the additive RDDs is The RDDKit. This is out of necessity as The RDDKit is a collection of six additive RDDs of which any or all can be used simultaneously. The six RDDs are shown in Figure 4.

So in the program you would write:

```
setInherit({"DBFNTX"})
USE STOCK VIA "CRYPT"
```

As has been mentioned above, CA-Clipper has no standard for inheritance, so The RDDKit's own `setInherit()` function is used. As usual, when a standard has not been defined it is difficult for third party vendors to maintain compatibility (eg FlexFile's `V_SuperRDD()` and The RDDKit's `setInherit()`).

However, as many RDDs can be used in the inheritance chain you could also write:

```
setInherit({"MEMO", "CRYPT", ;
            "DBFNTX"})
USE STOCK VIA "BINARY"
```

where BINARY inherits from MEMO which inherits from CRYPT which inherits from DBFNTX.

## Memo Fields

dBase's DBT memo fields are notoriously inefficient in all manner of ways. CA-Clipper's file compatible DBT memo fields are smarter, but still suffer from similar problems: the foremost being the dreaded memo bloat. For this reason different memo systems have been devised.

DBFSIX and DBFCDX both use FoxPro's more efficient FPT memo files. DBFSIX also has the ability to specify a different block size for the memo file which can further reduce memo bloat. FlexFile allows a host RDD's memo system to be replaced with Ganahl Software's proprietary DBV memo files. And, although dBxStore, is an API library and not an RDD, it uses DBX memo files.

Again the problem of physical existence rears its ugly head. CA-Clipper has no `memoExt()` function equivalent to its `indexExt()` function for indexes. DBFSIX has an `sx_MemoExt()` function but this is only appropriate if you use DBFSIX. As a result, any code which tests for the existence of memo files, copies memo files or erases memo files must have the memo file extension hard-coded.

## Weakly Typed Fields

Since the dawn of dBASE II, all fields in DBF files have been strongly typed. That is, a field could be guaranteed to have the same data type from one record to the next. With DBFSIX and FlexFile (and dBxStore if you count API libraries) this simple rule changed. Memo fields can have variable data types. So the code extract in Figure 5 is permissible.

Now it is very useful to be able to record 'NIL' instead of 'empty' (there is an important difference) and, depending on your attitude to normalisation, it may be useful to store many record values as an array instead of many records. However, this feature is only useful if your code has been written to be aware of it. Consider how much existing code has been written

### Guidelines for RDD Independence

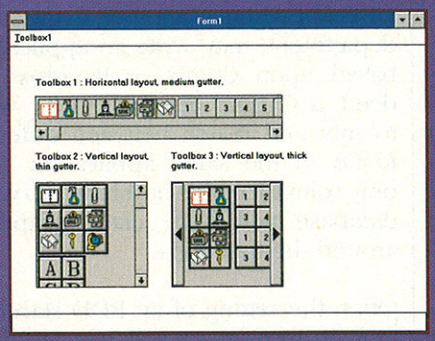
- Any code which is dependent upon the physical existence of tables should by wrapped in function calls so that it can easily be modified.
- Always use tag names instead of index order numbers to refer to index orders. Tag names are essential if the underlying RDD supports multiple order index bags as the relative orders inside the bag can change frequently so `order 3` will not always refer to the same index order.
- Use tag names which would be the same name as the index order bag name would have been if you had used single order index bags. So the first index order of the stock table could be given a tag name 'STOCK1' and the second order 'STOCK2'. This will enable you to switch to an RDD which doesn't have tag persistence (eg CA-Clipper V5.2 DBFNDX).
- Always specify a table default index order using a tag name. When multiple order bags are opened an index is not automatically set as it is for single order bags like DBFNTX (ie `SET ORDER TO 1`) so setting a default index order will assure easy transition between multiple order bags and single order bags.
- Index keys should be based upon fields only as not all RDDs support indexes built upon expressions (e.g. Paradox primary key indexes).
- Place primary key fields together and in order at the beginning of the field list as Paradox primary key indexes can only be based upon contiguous fields at the beginning of the field list.
- Use table lock methods as if multiple record locking is supported. That is, always pass the record identity to be locked. When you need to lock multiple child records use a manifest constant to determine whether to use multiple record locks:

  ```
  if MULTIRECLOCKS
    myLockRecordsInRange()
  else
    fLock()
  endif
  ```

  In this way you will be able to immediately reap the benefits of multiple record locking if you switch to a suitable RDD but your code will still work with single record lock RDDs.
- Never use the dbRecall() function as not all data sources can physically recall deleted records.
- `RecNo()` does not return a numeric value for all data sources as not all data sources have a concept of a record number. As a result, code which accepts record number parameters should simply compare the parameter against NIL instead of testing the data type using `VALTYPE()`.

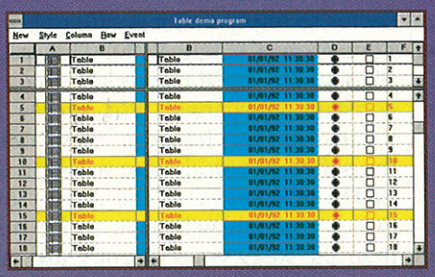
# 5 Great New Addons for Visual Basic in one package!

1



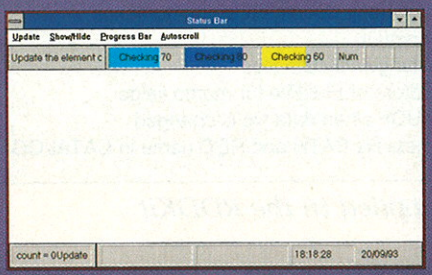

The toolbox is made up of groups of items. A toolbox can have any number of groups and each group can have any number of items.

2



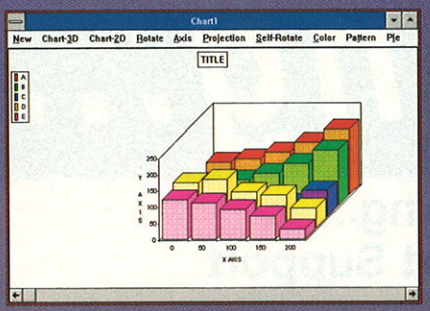

The table can be a regular table that accesses data from memory of it can be treated as a database front-end that maintains a cache of current data and fetches the data from the application only when needed.

3



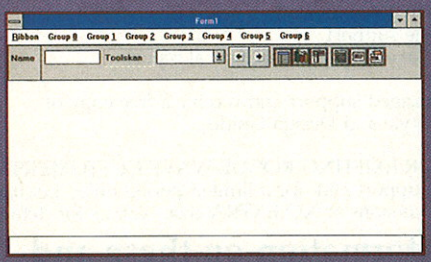

The status bar can have a maximum of 20 elements including progress meters, text and Separators.

4




The chart can be either 2D or 3D. It features: pie, line, bar, legends, labels, grid-lines, axis. The grid can also be configured to act as a database front-end that maintains a cache of a range of current data and fetches data from the application only when needed.

5




The Ribbon control is a horizontal bar that integrates combo boxes, edit controls, text strings and items (buttons)

# ToolsKan/VB

**£129.95**

+ Carriage + VAT

ExpoTech CORPORATION

ExpoTech House, Western Road, Hove, Sussex. BN3 1AF.
Sales: +44-(0)273-749222 Fax: +44-(0)273-746446

which is not expecting field data types to change from one record to the next.

## Sharing Code

There is a basic assumption made by third party vendors which is unfounded in some circumstances. The assumption is this: 'It is acceptable to achieve reuse of code by recompiling with different header files.'

The recompilation using different header files allows commands like:

```
SET SCOPETOP TO
SET SCOPEBOTTOM TO
SET FILTER TO
```

to be mapped onto different function calls (eg `cmxSetScope()` and `cmSetFilter()` instead of `sx_setScope()` and `m6_setFilter()`).

First, this assumption only holds if you always use commands and not functions. This may or may not be true. If you, like me, were enjoying the transition to a wholly function-based language and had ditched commands in favour of a more C-like style then this assumption is false.

However, there is a bigger barrier to the reuse of source code. That is when the source code is contained in a departmental library shared either by many programmers and/or shared by many applications. In this circumstance you cannot recompile the library. This is partly because you may not have access to the source code, but more likely because, if you recompile the library, you either recompile it for everyone else (and every application) or you create a separate version of the library. Multiple copies of a departmental library soon leads to disaster.

It is quite likely that one programmer may write an application based upon DBFNTX whereas another, in the same department, may write an application based upon Comix or Paradox. Indeed, a Clipper developer may want to mix and match between different RDDs in the same application. The only solution to this problem is to write database and table class wrappers around the functions.

Often the setting of an RDD standard (eg all application will use DBFSIX) seems to imply that it is safe to build a library on a single RDD. This trap is highlighted by the maxim: 'the only constant is change'. Put simply this means that as change is inevitable in the software world there will be a new RDD out soon (say 'The Acme Seven Driver') which will have to be better than the RDD you currently use. So if

| | | |
|---|---|---|
| BINARY | - | Long integer and double precision numerics |
| CRYPT | - | Data encryption |
| JOURNAL | - | Log all changes to data files |
| MEMO | - | Use dBxStore or FlexFile for memo fields |
| TRIGGER | - | Execute UDF when data file is changed |
| CATALOG | - | Lookup data file PATH and RDD name in CATALOG |

*Figure 4 - RDDs supplied in the RDDKit*

FarPoint

# A TRUE SPREADSHEET CONTROL!

**(THIS IS NOT A GRID!)**

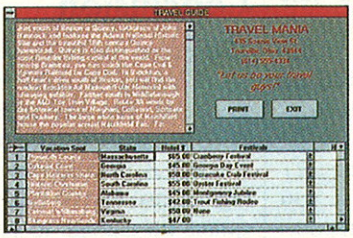
## Drover's Professional ToolBox for Windows

With 23 custom controls, including FarPoint's **industry unique full-featured spreadsheet control** (not a grid!), Formatted Edit Controls, Tool Bar and Status Bar, ability to add 3D effects to dialog boxes, View Pictures with animation, and Enhanced Listbox this package is a developer's dream come true.

Over 300 functions are built in, including DOS System functions, Date/Time support, String functions, and Enhanced file support.

Drover's Professional Toolbox supports MSC 7.0, Borland C++, Turbo Pascal, Actor, WindowsMaker Pro, Borland Resource Workshop and Dialog Editor.

Drover's Professional Toolbox for Windows is only £295, and of course there are no royalties!

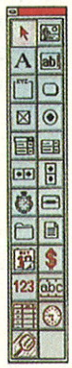



**DON'T CHEAT YOUR END USER WITH A GRID WHEN YOU COULD BE GIVING THEM A TRUE SPREADSHEET CONTROL.**

FarPoint's Spreadsheet control is unparalleled by any other package in its features, flexibility and power. The spreadsheet may be optionally locked so the user cannot make any changes. The width and height of columns and rows may be changed by the programmer or by the user. The font, color, and data type may be changed for any row, column or cell. Cells can have the following data types: edit, date, time, integer, float, static, formatted pic, combo box, button, and picture. Formulas can be added to any cell. Editing is performed within the cell.

## Visual Architect™ for Visual Basic

**"Visual Architect™ is a product that no serious Visual Basic developer can do without."**



Along with the industry's most powerful and flexible true Spreadsheet custom control (not a grid!), Visual Architect™ features Date with Calendar, Time, Float, Integer, Formatted PIC and View Text controls.

The Visual Architect™ manual is professionally written and contains extensive documentation.

Visual Architect is only £195, and of course there are no royalties!

A demo is available on CIX in the Farpoint conference.

Dealer enquiries welcome.

**To order please contact UK Distributor:**

ExpoTech (U.K.)
ExpoTech House, Western Road, Hove, Sussex, BN3 1AF
Tel: (0273) 749222 Fax: (0273) 746446

you change the standard to the new RDD all of the old applications based on the old RDD are unsupportable.

Much CA-Clipper DML (Data Manipulation Language) is genuinely polymorphic. Commands and functions like `SKIP`, `FOUND()` and `EOF()` are reusable no matter what the RDD or the circumstances. In theory, it should be possible to apply such commands independent of the actual data source. Additions made by third party RDDs (eg `SET SCOPE TO`) are often reusable if you have the luxury of being able to recompile with different header files. Some differences simply cannot be mapped (eg erasing an NTX file requires the NTX to be closed but erasing a tag requires the index to be open). Other differences require some forethought in the planning stage. I have provides some tips to code reuse in the box 'Guidelines for RDD independence'.

## Conclusion

The age of the RDD is definitely here. A lot of code is reusable from one RDD to the next, but the transition from one RDD to the next is far from seamless. Of the RDDs mentioned only The Advantage xBase Server DBFNTXAX RDD requires no change to your source. Some differences arise from the different types of physical existence; some from differences in syntax (eg `sx_setScope()` and `cmxSetScope()`) and some arise from conceptual differences (eg multiple order bags instead of single order bags). The only way to achieve closer to 100% RDD independence is through encapsulating all data access using table classes.

An interesting final point of note is that the forthcoming CA-Visual Objects For Clipper (aka Aspen) will support both RDDs and table classes. However, RDDs will only be available for record-oriented engines (eg dBASE, BTrieve, Paradox) whereas table classes will be available for both record- and set-oriented engines (Oracle, SQLBase, SQL Server).

EXE

```
* assume TEL_NO is a memo field
   GO TOP
   REPLACE TEL_NO WITH "081 994 6477"
   SKIP
   REPLACE TEL_NO WITH NIL
   SKIP
   REPLACE TEL_NO WITH {"081 994 6477", "081 994 4842", "0753 577733"}
   GO TOP
   ? valType(TEL_NO)    //   "C"
   SKIP
   ? valType(TEL_NO)    //   "NIL"
   SKIP
   ? valType(TEL_NO)    //   "A"
```

*Figure 5 - Variable data types in memo fields*

*Guy Smith is tired of the drastic reusability of OO code and the sub-second link lines of modern linkers and longs for the days of `SELECT PRIMARY` and `SELECT SECONDARY` and the five minute link times of Plink86.. NOT!*

*He can be contacted on 0602 865943 or 100014.1724 on CompuServe or smithg on CIX.*

# Save My Screen!

*Far from being some kind of Windows black magic, screen savers are simply plain vanilla executables. Bruce Forman reveals how to write one.*

Screen savers are often thought of as one of those amusing 'cubby-holes' that users new to Windows explore, along with wallpaper and bitmaps. However, apart from their ability to provide 'factor 20' protection against 'screen-burn' they have a more powerful role to fulfil in the area of PC security.

For a standalone PC, the protection provided is restricted to preventing the casual passer-by from meddling with whatever is running at that time. Unless you have a machine with a built-in BIOS password, the determined hacker will merely reboot to gain access to data on your hard disk. Where network machines are concerned, a password protected screen saver can provide a high level of protection if you need to leave your PC for any time. An unattended machine which is logged into a network can be a serious security problem: but you don't always want to logout every time you leave your desk to grab a cup of caffeine.

## Securing your PC

As you are no doubt aware, the screen savers shipped with Windows can be accessed via the DESKTOP utility in Control Panel. You are able to associate a password with the screen saver you have chosen. At the same time you can also change the period of keyboard or mouse inactivity after which time, Windows causes the screen saver to run. The only problem with this method of running the screen saver is that, if you set a reasonable time period of say 5 minutes to prevent the screen saver cutting in too often, then this will still present a security problem for that period of time until the screen saver is run.

I wanted to be able to protect my PC by clicking on an icon to run the screen saver immediately. It turns out that this can be achieved quite easily since Windows screen savers are all, in fact, executable files in their own right. If you look in your Windows directory you should find several files with the .SCR extension. These are the screen saver files; to execute them directly you should rename them to .EXE files.

```
Declare Function ShowCursor% _
  Lib "User" (ByVal bShow%)

Declare Function GetPrivateProfileString% _
  Lib "Kernel" (ByVal lpApplicationName$, _
                ByVal lpKeyName$, _
                ByVal lpDefault$, _
                ByVal lpReturnedString$, _
                ByVal nSize%, _
                ByVal lpFilename$)

Declare Function WritePrivateProfileString%
  Lib "Kernel" (ByVal lpApplicationName$, _
                ByVal lpKeyName$, _
                ByVal lpString$, _
                ByVal lplFileName$)

Global gbOldPasswd$
Global gbNewPasswd$
Global gbDrawBoxSpeed%

Sub Encrypt (uncoded$, coded$, keystring$)
  newkeystring$ = ""
  For i = Len(keystring$) To 1 Step -1
    newkeystring$ = newkeystring$ + _
      Mid$(keystring$, i, 1)
  Next
  length = Len(uncoded$)
  coded$ = ""
  For i = 1 To length
    letter = i - Len(keystring$) * _
             Int(i / Len(keystring$) - .1)
    newascval = Asc(Mid$(newkeystring$, _
                letter, 1)) + _
                Asc(Mid$(uncoded$, i, 1))
    coded$ = coded$ + Chr$(newascval)
  Next
End Sub

Sub Main ()
  'Prevent multiple instances of
  ' program running
  If App.PrevInstance = True Then
    Exit Sub
  End If
  Randomize

  'INI file is in application's directory
  lpFilename$ = App.Path
  If Right$(lpFilename$, 1) <> "\" Then
    lpFilename$ = lpFilename$ + "\"
  End If
  lpFilename$ = lpFilename$ + "BOXES.INI"

  ' Create INI file if one does not exist
  If Len(Dir$(lpFilename$)) = 0 Then
    Open lpFilename$ For Output As #1
    Print #1, "[Boxes Screen Saver]"
    Print #1, "Password="
    Print #1, "DrawBoxSpeed=1"
    Close #1
  End If

  'Setup parameters for reading password
  ' from INI file
  lpApplicationName$ = "Boxes Screen Saver"
  lpKeyName$ = "Password"
  lpDefault$ = ""
  lpReturnedString$ = Space$(81)
  nSize = 81

  'Read password and unencrypt
  n% = GetPrivateProfileString% _
  ( _
    lpApplicationName$, _
    lpKeyName$, _
    lpDefault$, _
    lpReturnedString$, _
    nSize, _
    lpFilename$ _
  )
  lpReturnedString$ = _
    Left$(lpReturnedString$, _
          InStr(lpReturnedString$, _
                Chr$(0)) - 1)
  UnEncrypt lpReturnedString$, uncoded$, _
            "enigma"
  gbOldPasswd$ = uncoded$

  'Setup parameters for reading
  ' DrawBoxSpeed from INI file
  lpKeyName$ = "DrawBoxSpeed"
  lpDefault$ = ""
  lpReturnedString$ = Space$(81)
  nSize = 81

  'Read DrawBoxSpeed from INI file
  n% = GetPrivateProfileString% _
       ( _
         lpApplicationName$, _
         lpKeyName$, _
         lpDefault$, _
         lpReturnedString$, _
         nSize, _
         lpFilename$ _
       )
  lpReturnedString$ = _
    Left$(lpReturnedString$, _
          InStr(lpReturnedString$, _
                Chr$(0)) - 1)
  gbDrawBoxSpeed% = Val(lpReturnedString$)

  'Show Setup Window or Screen Saver
  ' depending on command line parameter
  Select Case Command$
  Case "/c", "/C"
    'Show mouse pointer
    X% = ShowCursor(True)
    Setup.Show
    'Hide mouse pointer
    X% = ShowCursor(False)
  Case "/s", "/S"
    ScreenSaver.Show
  End Select
End Sub

Sub UnEncrypt (coded$, uncoded$, keystring$)
  newkeystring$ = ""
  For i = Len(keystring$) To 1 Step -1
    newkeystring$ = newkeystring$ + _
                    Mid$(keystring$, i, 1)
  Next
  length = Len(coded$)
  uncoded$ = ""
  For i = 1 To length
    letter = i - Len(keystring$) * _
             Int(i / Len(keystring$) - .1)
    newascval = Asc(Mid$(coded$, i, 1)) _
                - Asc(Mid$(newkeystring$, _
                           letter, 1))
    uncoded$ = uncoded$ + Chr$(newascval)
  Next
End Sub
```

*Figure 1 - Screen Saver main module*

# Multi-Platform 32-bit Power: WATCOM C/C++32

- Extensive C and C++ Support
- The Widest Range of 32-bit Intel x86 Platforms
- The Industry's Leading Code Optimizer
- Multi-platform, Cross-development Toolset

## Professional C and C++ Development Tools

C/C++32 delivers the key technologies for professional developers: comprehensive C++ support including templates and exception handling; advanced superscalar optimization; and 32-bit multi-platform support. C/C++32 includes both C and C++ compilers, so you can incrementally adopt the benefits of C++.

## Unleash 32-bit Power!

C/C++32 delivers 32-bit performance. The 32-bit flat memory model simplifies memory management and lets applications address beyond the 640K limit. Powerful 32-bit instruction processing delivers a significant speed advantage: typically a minimum 2x processing speedup.

## A C++ compiler designed to deliver on the promise of object-oriented programming

The C++ compiler provides comprehensive support for the AT&T version 3.0 language including templates, plus exception handling. These features are key to realizing the benefits of object-oriented programming: code reusability, increased reliability and reduced maintenance.

## Hot New Superscalar Code Optimizer

The hot, new C/C++32 code generator advances the performance envelope. New superscalar optimization strategy uses "riscification" and instruction scheduling to deliver improved performance on 486 and Pentium processors. The compiler can create a single, high-performance executable which runs on 386, 486 and Pentium processors.

## Industry Standard. Industry's Choice

WATCOM's working relationships with industry leaders such as Autodesk, GO, IBM, Intel, Lotus, Microsoft and Novell ensure that we continue to understand and meet the needs of the software industry and professional developers.

## Multi-Platform, Cross Development Support

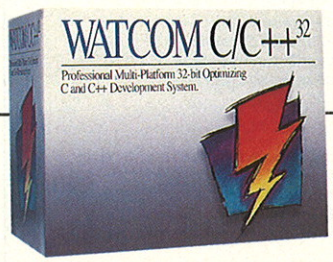
C/C++32 supports a wide range of 32-bit Intel x86 host and target platforms allowing professional developers to leverage the multi-platform, cross-development capabilities of today's operating environments including OS/2 2.x and Windows NT.

- **32-bit DOS** host and target support; includes the DOS/4GW 32-bit DOS extender by Rational Systems with royalty-free run-time and virtual memory support up to 32Mb.
- **OS/2 2.x** host and target support; enables the development of OS/2 2.x applications and DLLs. 32-bit Windows 3.x applications can exploit WIN-OS/2 using WATCOM's Windows Supervisor technology.
- **Windows NT** host and target support; includes a Windows NT-hosted debugger for 16-bit Windows 3.x, Win32s and Windows NT debugging; also includes **Win32s** target support.
- **32-bit Windows 3.x** target support; includes WATCOM's 32-bit Windows Supervisor, and enables development and debugging of true 32-bit Windows 3.x applications and DLLs.

### Also Available: WATCOM C32 for DOS

C32 for DOS is a professional, low cost 32-bit C compiler and tools package enabling development, debugging, performance profiling and royalty-free distribution of 32-bit applications for extended DOS.
**Suggested retail price: £139**

**WATCOM C/C++32** has a suggested retail price of **£419***
For additional information or to order direct call 0800 44 44 55.
Call our FAX Back system at 0101-519-747-2693 from you fax machine for immediate product information.




WATCOM 0800 44 44 55

## The Leader in 32-bit Development Tools

PO Box 64, Livingston, West Lothian, EH54 7AE, UK Telephone: 0506 460112, Fax: 0506 460115

*Price in Pound Sterling. Does not include freight and VAT where applicable. Authorised dealers may sell for less.
WATCOM C and the Lightning Device are trademarks of WATCOM International Corp. DOS/4G and DOS/16M are trademarks of Rational Systems Inc. Other trademarks are the properties of their respective owners.

If you now create a program item with the new executable filename, you will end up with an icon which, when run, will start the screen saver. You should note, however, that there are two command line parameters associated with the new executable. The /S parameter is necessary to start the screen saver. The /C option will give access to the screen saver's setup window.

## The ingredients

Creating our own screen saver involves following a few simple procedure: from thereon we are confined only by the limits of our imagination. Windows screen savers usually consist of four main elements:-

- The graphical display which constitutes the screen saver itself.
- A setup window for altering the operating parameters of the screen saver.
- A password changing window and routines.
- A password prompt window.

When a typical Windows screen saver, such as Starfield Simulation, is activated, it will run until either a key is pressed on the keyboard, or the mouse is moved. Assuming that a password is set, then the user will be prompted for the password. And if the correct one is entered, the user will be returned to his Windows desktop. If an incorrect password is entered then the screen saver will resume until there is further keyboard/mouse activity.

The setup window allows the user to set operating parameters of the screen saver and a password if required. In the case of Starfield Simulation, the operating parameters which can be altered are the speed of movement through the stars and the density of the starfield.

## A Program Example

The example that I have created contains only a very simple screen saver, written for Visual Basic for Windows (VBW), which draws rectangles of random size, colour and position on the screen. All the forms and procedures discussed are available on diskette. See end of article for details. Some of the procedures involved, such as password input and reading/writing information strings from .INI files, have wider uses. So, even if you are not interested in creating screen savers, then I hope you may find these useful.

My screen saver is called Boxes. It uses four forms - one for each of the elements described above - and a module of common procedures.

## General Module

The code in Figure 1 contains the declarations for three Windows API calls which are used in BOXES and three procedures. The subroutine `Main()` is set as the `Startup` form (by selecting

```
Sub chkPassword_Click ()
  If chkPassword.Value Then
    cmdSetPassword.Enabled = True
  Else
    cmdSetPassword.Enabled = False
  End If
End Sub

Sub cmdCancel_Click ()
  Unload Setup
End Sub

Sub cmdOK_Click ()
  If chkPassword.Value = 1 Then
    If Len(gbNewPasswd$) Then
      Encrypt gbNewPasswd$, coded$, "enigma"
      lpReturnedString$ = coded$
    Else
      Encrypt gbOldPasswd$, coded$, "enigma"
      lpReturnedString$ = coded$
    End If
  Else
    lpReturnedString$ = ""
  End If

  gbDrawBoxSpeed% = _
    Val(txtDrawBoxSpeed.Text)
  If gbDrawBoxSpeed% < 1 Then _
    gbDrawBoxSpeed% = 1
  If gbDrawBoxSpeed% > 10 Then _
    gbDrawBoxSpeed% = 10

  lpFilename$ = App.Path
  If Right$(lpFilename$, 1) <> "\" Then _
    lpFilename$ = lpFilename$ + "\"
  lpFilename$ = lpFilename$ + "BOXES.INI"
  lpApplicationName$ = "Boxes Screen Saver"
  lpKeyName$ = "Password"
  Result% = _
    WritePrivateProfileString% _
    ( _
      lpApplicationName$, _
      lpKeyName$, _
      lpReturnedString$, _
      lpFilename$ _
    )
  lpKeyName$ = "DrawBoxSpeed"
  lpReturnedString$ = _
    LTrim$(Str$(gbDrawBoxSpeed%))
  Result% = _
    WritePrivateProfileString% _
    ( _
      lpApplicationName$, _
      lpKeyName$, _
      lpReturnedString$, _
      lpFilename$ _
    )
  Unload Setup
End Sub

Sub cmdSetPassword_Click ()
  SetPassword.Show
End Sub

Sub Form_Load ()
  Setup.Left = _
    (Screen.Width - Setup.Width) / 2
  Setup.Top = _
    (Screen.Height - Setup.Height) / 2
  If Len(gbOldPasswd$) Then
    chkPassword.Value = 1
    cmdSetPassword.Enabled = True
  End If
  txtDrawBoxSpeed.Text = _
    LTrim$(Str$(gbDrawBoxSpeed%))
End Sub

Sub spnDrawBoxSpeed_SpinDown ()
  gbDrawBoxSpeed% = gbDrawBoxSpeed% - 1
  If gbDrawBoxSpeed% < 1 Then _
    gbDrawBoxSpeed% = 1
  txtDrawBoxSpeed.Text = _
    LTrim$(Str$(gbDrawBoxSpeed%))
End Sub

Sub spnDrawBoxSpeed_SpinUp ()
  gbDrawBoxSpeed% = gbDrawBoxSpeed% + 1
  If gbDrawBoxSpeed% > 10 Then _
    gbDrawBoxSpeed% = 10
  txtDrawBoxSpeed.Text = _
    LTrim$(Str$(gbDrawBoxSpeed%))
End Sub

Sub txtDrawBoxSpeed_KeyPress _
( _
  KeyAscii As Integer _
)
  If KeyAscii = 13 Then
    gbDrawBoxSpeed% = _
      Val(txtDrawBoxSpeed.Text)
    If gbDrawBoxSpeed% < 1 Then _
      gbDrawBoxSpeed% = 1
    If gbDrawBoxSpeed% > 10 Then _
      gbDrawBoxSpeed% = 10
    txtDrawBoxSpeed.Text = _
      LTrim$(Str$(gbDrawBoxSpeed%))
  End If
End Sub
```

*Figure 2 - Setting parameters*

```
Sub cmdCancel_Click ()
  gbNewPasswd$ = ""
  Unload SetPassword
End Sub

Sub cmdOK_Click ()
  If Len(gbOldPasswd$) Then
    If txtOldPassword.Text <> gbOldPasswd$ _
    Then
      MsgBox "Incorrect old password", 48, _
             "Error"
      txtOldPassword.Text = ""
      txtNewPassword.Text = ""
      txtRetypePassword.Text = ""
      txtOldPassword.SetFocus
      Exit Sub
    End If
  End If
  If Len(txtNewPassword.Text) = 0 Then _
    Exit Sub
  If txtNewPassword.Text <> _
  txtRetypePassword.Text Then
    MsgBox "New passwords do not match",
48, _
           "Error"
    txtNewPassword.Text = ""
    txtRetypePassword.Text = ""
    txtNewPassword.SetFocus
    Exit Sub
  Else
    gbNewPasswd$ = txtNewPassword.Text
    Unload SetPassword
  End If
End Sub

Sub Form_Load ()
  If Len(gbOldPasswd$) = 0 Then
    lblOldPassword.Enabled = False
    txtOldPassword.Enabled = False
    lblNewPassword.Enabled = True
    txtNewPassword.Enabled = True
    lblRetypePassword.Enabled = True
    txtRetypePassword.Enabled = True
  End If
End Sub

Sub txtOldPassword_Change ()
  lblNewPassword.Enabled = True
  lblRetypePassword.Enabled = True
  txtNewPassword.Enabled = True
  txtRetypePassword.Enabled = True
End Sub
```

*Figure 3 - Changing passwords*

# System Science

• Software • Specialists • Software • Specialist • Software • Specialists • Specialist • Software • Specialists •

## UNIX

**SCO** Unix Op. Sys **SCO** Development Sys.
**SCO** Open Desktop - NEW 3.0 **SCO Xsight**
**SCO** TCP/IP & NFS **FTP** PC/TCP
**Interactive** Unix Op System/ Architect Series
**Informix** 4GL & SQL **RM** Cobol-85
**LPI** Fortran **LPI C++** SCO, SUN, HPUnix, etc
**C++ Views** for Motif - rapid portable GUI dev
**Word Perfect** for Unix **PC-NFS**
**Zinc** Interface **Framework** for Motif
... many more for 386 Unix and Sun, HP, DEC Risc etc

## Mathematics

**Derive** - (plus free Intro book)
**Derive XM** - NEW for 386 extended memory
**Mathematica 2.2** Win, MAC, Workstation, Student Ver
**MathCad** Windows **Lindo** Linear Prog
**Origin** Graphing with 3D & Contour for Windows
**Gauss** **Systat for Windows**
**Mathtype Win** **PC Matlab for Win**

## Windows Development

(see Microsoft , Borland, Zortech, Watcom)

| | |
|---|---|
| **Data Entry Workshop** by Turbopower | £169.00 |
| **Win /Sys Lib** | £99.00 |
| **HALO** Imaging Lib Windows | £395.00 |
| **Smalltalk/V** Windows | £285.00 |
| PCX ProgToolkit Windows | £169.00 |
| CASE:W & KASE:W VIP | £call |
| INSTALLShield ver 2.0 | £265.00 |
| **Plug-In** Edit Control | £375.00 |
| **WinGen** - screen generator | £125.00 |
| **Doc-To-Help** for Help file prep | £call |

## CrossDevelopment

| | |
|---|---|
| **2500AD** Cross Assemblers | £175.00 |
| **Hitech C Cross** Z80, 6805, 68HC11.. | £360.00 |
| **Introl C** or Mod-2 Cross Comp-6809, 68K Aztec C68K, Avocet CCross | £call |

## VB Value

| | |
|---|---|
| **Visual Architect** | £195.00 |
| **Q&E Database/VB** | £135.00 |
| **QuickPak Prof for Windows** | £149.00 |
| **3D Gizmos** from MicroHelp | £79.00 |
| **CommLib/Win** from Microhelp | £110.00 |
| **Network Lib/Win** from MicroHelp | £79.00 |

## For Fortran

| | |
|---|---|
| **Lahey F77L-EM/32** new faster 5.1 | £765.00 |
| **Graphoria for F77L-EM/32** | £245.00 |
| **Salford FTN 77** Developer | £745.00 |
| **PC/Interacter** (scrns & graphics) | £325.00 |
| **Halo Professional for Fortran** | £245.00 |
| **Ingraf** - with source | £245.00 |
| **RM Fortran** | £375.00 |

## Microsoft

| | |
|---|---|
| Visual C/C++ - 32 bit Edition *NEW* | £299.00 |
| Visual C/C++ PE - disk or CD-ROM | £call |
| Delta - Windows Version Control | £call |
| MultiMedia Viewer Pub Kit | £245.00 |
| MS Test for Windows - *New ver 2.0* | £245.00 |
| MS Windows DDK 3.1 | £275.00 |
| MS Macro Assembler 6.1 | £99.00 |
| Visual Basic - Windows 3.0 SE | £99.00 |
| Visual Basic Windows 3.0 PE | £229.00 |
| MS Fortran 5.1 | £115.00 |
| Visual Basic for MS-DOS | £99.00 |
| Visual Basic for MS-DOS Prof | £225.00 |
| MS Windows, Win SDK & WfWG | £call |

*upgrade to **Visual C/C++ & new Visual Basic 3.0***

***Authorised Languages Dealer***

## Borland

| | |
|---|---|
| Borland C++ & AFW **3.1** | £345.00 |
| *with free ProtoGen and Brief until 31 Oct '93* | |
| Borland C++ for OS/2 2.0 NEW | £call |
| Turbo C++ Visual Edition | £69.00 |
| Turbo C++ DOS & C++ Video | £85.00 |
| Turbo Pascal 7 | £85.00 |
| Borland Pascal with Objects | £225.00 |
| Paradox Engine - *ver 3.0* | £119.00 |

***Authorised Languages Dealer***

## Nu-Mega

| | |
|---|---|
| **Bounds Checker 2.0** for DOS | £159.00 |
| **BoundsCHECKER/W** ***NEW 2.0*** | £195.00 |
| **Soft-ICE/W (Windows)** | £295.00 |
| **Soft-ICE (for 386 DOS)** | £295.00 |

*Special pricing on Nu-Mega combi packs*

## Intersolv *(Authorised)*

| | |
|---|---|
| **PVCS Version Control** ***new 5.1*** | £345.00 |
| new WINDOWS version, call for upgrades | |
| multi-user for DOS Networks, OS/2 , Sun, HP9000, RS6000 | |
| **PVCS Config Builder** better make than Make | £175.00 |
| **PVCS Reporter for Win or OS/2** | £125.00 |

## MKS

| | |
|---|---|
| **MKS Toolkit** new 4.1 *with UUCP & AWKC* | £195.00 |
| **MKS RCS** - NEW Windows Ver | £call |
| **MKS Lex & Yacc** *for C, C++ & TP* | £195.00 |

## Lifeboat

| | |
|---|---|
| Dan Bricklin's Demo II - prototyping, demos etc | £175.00 |
| Advantage Disassembler | £195.00 |
| Essential COMMs & Graphics tkts | £call |

## Tools and Editors, Etc

| | |
|---|---|
| CodeWright Editor for Windows | £175.00 |
| Personal REXX | £95.00 |
| Kedit (Xedit for PC) ver 5 | £110.00 |
| **Select Case** - Yourdon or SSADM | £495.00 |
| **EasyCase** Plus Developer WIN ***NEW*** | £999.00 |
| **MetaDesign** - Flowcharts, DFDs etc | £295.00 |
| PC Logo | £60.00 |

## WATCOM

| | |
|---|---|
| **C/C++** $^{32}$ - new for DOS32, Win,OS/2 , ADS etc | £395.00 |
| **C /C++**- $^{16}$- new DOS & WIN | £325.00 |
| **Fortran 77** $^{32}$ - NEW for dos32,Win, ADS etc | £395.00 |
| **SQL** Dev Edition - Windows or DOS | £275.00 |
| **VX*REXX** visual REXX for OS/2 | £call |

## C/C++

| | |
|---|---|
| **Symantec C++** DOS , Win, 32 - *NEW* | £call |
| **Zortech C++ OS/2 2.0** - ***NEW 3.1*** | £175.00 |
| **Metaware High C/C++ 386** | £call |
| **Think C for MAC** - new ver 6.0 | £139.00 |

### C++ Libs

| | |
|---|---|
| **Rogue Wave C++Libs** | £call |
| **C++/Views** - Win, OS/2, Motif | £call |
| **Zinc** Interface - free training video | £call |
| **Poet** (OOPS Database) | £call |
| **Comm++** by Greenleaf - NEW Ver | £189.00 |

### C Datafile & Comms

| | |
|---|---|
| CodeBase 5.0 /CodeBase++ | £245.00 |
| Btrieve - DOS or Windows ver | £395.00 |
| Ctree Plus from Faircom | £call |
| Essential Comms | £210.00 |
| Greenleaf CommLib - 4.0 for DOS/Win | £235.00 |
| C Asynch Manager (Blaise) | £135.00 |

### C Screens & Graphics

| | |
|---|---|
| CScape with Look & Feel - NEW 4.0! | £345.00 |
| Panel Plus II (now withWin support) | £275.00 |
| Greenleaf DataWindows - DOS | £235.00 |
| Figaro+ - PHIGS for 386 | £745.00 |
| GX Graphics | £169.00 |
| GIF Programmer's Toolkit | £169.00 |
| GX Printer - NEW | £225.00 |
| HaloProfessional | £245.00 |
| MetaWindows 4/32 - NEW | £call |

### Misc. Libs

| | |
|---|---|
| Install Professional | £245.00 |
| C Tools Plus | £110.00 |
| Bar Code Library- source | £375.00 |

... many more tools

## Pharlap

| | |
|---|---|
| TNT DOS-Extender SDK | £365.00 |
| **286** DOS-Extender SDK | £365.00 |
| **386 ASM/Link/LOC** | £975.00 |

## Databases

| | |
|---|---|
| **Access** by Microsoft | £295.00 |
| **Fox Pro** - DOS or Windows | £call |
| **Paradox** - DOS or Windows | £call |
| **Crystal Reports** for Windows | £145.00 |

## What's New!

| | |
|---|---|
| **BoundsCHECKER/W ver 2.0** | £call |
| **and with MS TEST for Windows** | £special |
| **Symantec C/C++ DOS & Win - special offers** | |
| **Visual C/C++ - 32 bit Win32 & Win/NT** | £299.00 |
| upgrade from MS VC/C++Prof until 31 Oct 1993 | |
| **MS Multimedia Viewer Publisher Toolkit** | |
| **Watcom VX*REXX for OS/2** | £call |
| **EasyCASE System Designer WIN** | £1045.00 |

- **Prices are exclusive of VAT.**
- **Prices (except upgrades & specials) include delivery to mailand UK.**
- **Prices are subject to change.**
- **VISA, ACCESS and Mastercard welcome with telephone orders. E&OE**


**3-5 Cynthia St**
**London N1-9JF**
**Fax: 071 837 6411**

**071 833 1022**


• Software • Specialists • Software • Specialist • Software • Specialists •

the Project item of the Options menu in VBW). Consequently, it is the first procedure to run. Main() first checks if the program is already running: if not, it reads setup information from the BOXES.INI file. This is achieved using the Windows API call GetPrivateProfileString%(). If BOXES.INI does not exist in the directory where the program is being run, then it is created with default settings.

The password is then unencrypted using the subroutine UnEncrypt(). Depending on the command-line parameter, the program either runs the screen saver or presents the user with the setup window.

## Setting Parameters

Due to the simple nature of our screen saver I have presented only one setup parameter to control the speed at which the boxes are drawn (gbDrawBoxSpeed%). The code for my screen saver setup form is given in Figure 2. For the purposes of the example I have used a spin button in conjunction with a text box for setting the speed parameter which is used to prime a timer interval on SAVER.FRM to control how fast the boxes will be drawn. For those of you who do not have VBW Professional, the text box alone will suffice. I have restricted the permissible input range to be between 1-10.

I have also provided a way for a user to set a password using the Change Password dialog. The new password is encrypted using the Encrypt() subroutine in MODULE1.BAS (see Figure 3). This is then written along with the global gbDrawBoxSpeed% to the file BOXES.INI using the Windows API call WritePrivateProfileString%().

## Drawing Boxes

The screen saver form is setup at design time with a single timer control. When the form loads, the Windows API function ShowCursor() is used to hide the cursor. The original state of the cursor is taken so that it can be restored when the screen saver is terminated.

The timer interval is set according to the value of the setup parameter gbDrawBoxSpeed%. At the heart of my screen saver implementation is the event Timer1_Timer() which draws a rectangle every time count reaches the preset interval. To create your own screen savers, simply customise the behaviour in this routine.

## Unloading

When a key is pressed or the mouse is moved, the GETPASS.FRM is loaded, prompting the user to type in a password (so long as the screen saver has been set up with password enabled). If entered correctly, the screen saver is terminated. The VBW event which causes this response is Form_MouseMove() in Figure 4. However, to prevent the screen saver being unloaded as soon as it's loaded, the first instance of Form_MouseMove() must be ignored - and the mouse position checked to see if it has really moved. This is achieved by storing the x-y coördinates of the mouse position in static variables and exiting when these values are non-zero, or when they change.

If the password is correct, the screen saver is terminated and the user is returned to the Windows desktop. A timer control on this form (enabled when GETPASS.FRM is loaded), unloads the password prompt window after 10 seconds if no input is forthcoming (see Figure 5).

## Creating an EXE

Depending on how you wish to run your screen saver you should compile the program using the Make EXE File option in VBW's File menu, specifying a .SCR or .EXE file extension to the filename. A .SCR file should be placed, along with the other screen savers, in your Windows system directory. Use the DESKTOP utility in Control Panel to enable your screen saver.

## Passwords

One difficulty you should be aware of when writing screen savers in VBW concerns passwords. A password set using the standard Windows screen savers is not directly accessible from VBW. This is because the password is located in CONTROL.INI, but it is in an encrypted form (different to the encryption routines that I have employed). The procedures that Windows provides for gaining access to the password are in a static link library SCRNSAVE.LIB which is not accessible from VBW. I have seen

```
Dim CursorCount%

Sub Form_KeyDown _
(
  KeyCode As Integer, _
  Shift As Integer _
)
  If Len(gbOldPasswd$) Then
    GetPassword.Show 1
  Else
    Unload ScreenSaver
  End If
End Sub

Sub Form_Load ()
  'Record original mouse-pointer show count
  CursorCount = ShowCursor(False) + 1

  'Hide mouse pointer
  Do While ShowCursor(False) >= -1
  Loop
  Do While ShowCursor(True) < -1
  Loop

  Timer1.Interval = 1000 / gbDrawBoxSpeed%
End Sub

Sub Form_MouseMove _
(
  Button As Integer, _
  Shift As Integer, _
  X As Single, _
  Y As Single _
)
  Static Xlast, Ylast

  'Get current position in
  ' same variable types
  Xnow = X
  Ynow = Y

  'On first move simply record position
  If Xlast = 0 And Ylast = 0 Then
    Xlast = Xnow
    Ylast = Ynow
    Exit Sub
  End If

  'Quit only if mouse actually
  ' changes position
  If Xnow <> Xlast Or Ynow <> Ylast Then
    If Len(gbOldPasswd$) Then
      GetPassword.Show 1
    Else
      Unload ScreenSaver
    End If
    Xlast = 0
    Ylast = 0
  End If
End Sub

Sub Form_Unload (Cancel As Integer)
  'Restore original mouse-pointer
  ' show count
  Do While ShowCursor(False) >= CursorCount
  Loop
  Do While ShowCursor(True) < CursorCount
  Loop
End Sub

Sub Timer1_Timer ()
  x1% = Int(Rnd * Screen.Width)
  y1% = Int(Rnd * Screen.Height)
  x2% = Int(Rnd * Screen.Width)
  y2% = Int(Rnd * Screen.Height)
  colornum% = Int(16 * Rnd)
  ScreenSaver.Line (x1%, y1%)-(x2%, y2%), _
    QBColor(colornum%), BF
End Sub
```

*Figure 4 - The screen saver*

```
Sub Form_Load ()
  GetPassword.Left = _
    (Screen.Width - GetPassword.Width) / 2
  GetPassword.Top = _
    (Screen.Height - GetPassword.Height) / 2
  Timer1.Enabled = True
  Timer1.Interval = 10000
End Sub

Sub Form_Unload (Cancel As Integer)
  Timer1.Enabled = False
End Sub

Sub Timer1_Timer ()
  Unload GetPassword
End Sub

Sub txtPassword_KeyPress _
(
  KeyAscii As Integer _
)
  If KeyAscii = 27 Then Unload GetPassword
  If KeyAscii = 13 Then
    If GetPassword.txtPassword.Text = _
    gbOldPasswd$ Then
      Unload ScreenSaver
    End If
      Unload GetPassword
  End If
End Sub
```

*Figure 5 - Verifying a correct password*

# Clipper Developers


Competitive £129.00 Upgrade!

**When you need a function...**

1989 DATA BASED ADVISOR Readers Choice Award

1990 DATA BASED ADVISOR Readers Choice Award

1991 DATA BASED ADVISOR Readers Choice Award

1992 DATA BASED ADVISOR Readers Choice Award

# Get FUNCky!

The FUNCky add-in library for CA-Clipper contains over 1000 functions which can be used as if they were a direct part of the Clipper language. Extensive features include extraordinary Low Level file handling, blazing fast String Manipulation and impressive Video control. Graphics, Mouse, Arrays, Disk, Text Editor, Get System Replacement, Object File Librarian, C compatibility, 90 days free technical support and so much more in one easy to use well documented package.

Get FUNCky now and get the phenomenal programming power you need to create advanced applications demanded of you in today's competitive development environment. Free, fully working demo disks, and full function lists are available for your approval. A money back guarantee ensures a no risk trial period and our Competitive upgrade can make you a part of the FUNCky family at the lowest prices ever! There has never been a better time to get FUNCky, make that call!

**Order Today!**

FUNCky™
LIBRARY DEVELOPMENT SYSTEM

QBS

QBS SOFTWARE LTD
10 Barley Mow Passage, Chiswick,
London W4 4PH
Tel: 081 994 4842 Fax: 081 994 3441
BBS: 081 747 1979

dynamic link libraries written by third parties which will provide this access from VBW - but these cost money for distribution rights.

There are two work-arounds. Either write your own encryption routine, as I have done in my example screen saver, or alternatively, write your own DLL for calling these Windows procedures.

The encryption routines that I have used were cribbed from a magazine article some time ago on encryption techniques. Unfortunately I can't remember which magazine it was, so I'm afraid I cannot provide a reference. Apologies to the author responsible for the original article. The routines work on the basis of a known keystring which is available to both the `UnEncrypt()` and `Encrypt()` procedures.

## Final Thoughts

As I said above the actual screen saver graphics in the example are fairly routine. For those of you now fired with enthusiasm for writing your own screen saver, the limits are your own imagination.

The method I have outlined is only a framework for producing screen savers using VBW, but there *are* other ways. I'm exploring some of the Windows API calls to try to produce a richer set of graphics displays using things like bit-block transfers.

***You don't always want to logout every time you leave your desk to grab a cup of caffeine***

I hope the techniques that I have presented here have been interesting and informative. If nothing else, I strongly urge those of you working in a networked Windows environment to look at a password protected screen saver as a way of increasing the security of your network nodes, when they are left unattended.

EXE

---

*Bruce Forman is a Network Supervisor at the University of East London and runs an independent consultancy offering programming services. He can be reached via email at bruce@uk.ac.uel.bkmain.*

*To obtain a copy of the complete Visual Basic source for the screen saver described by Bruce, please follow the instructions given in Column 1 of the Contents page. Mark your envelope 'SCNSAV'.*

**References**

*Visual Basic Programmer's Guide to the Windows API*,
Daniel Appleman, Ziff-Davis Press,
ISBN 1-56276-073-4

*Microsoft Visual Basic Workshop*,
John Clark Craig, Microsoft Press,
ISBN 1-55615-512-3

# Put an end to software piracy!

Meet the growing family of security keys from Software Security.

Each one a specialist at enforcing your license agreement in virtually any user environment you can think of. Whether it's DOS, UNIX, Macintosh or OS/2... whether it's a single user installation or a LAN.

Simply connect the appropriate key to a single user computer, or a non-dedicated file server in a network, and you control all access to your protected application.

Users, however, won't even know it's there. The keys are transparent and won't impact software functionality or the ability to make back up copies. Normal node and LAN operations are unaffected.

Simple. Unassuming. Ever vigilant. Easy to incorporate into your application package. And quite possibly the most profitable hardware investment a software developer can make.

**To find out more, call:**

**(0784) 430060**

## THE ULTIMATE PROTECTION FOR SOFTWARE PUBLISHERS




**Software Security International Ltd**

**21a The Precinct, High Street, Egham, Surrey, United Kingdom, TW20 9HN**

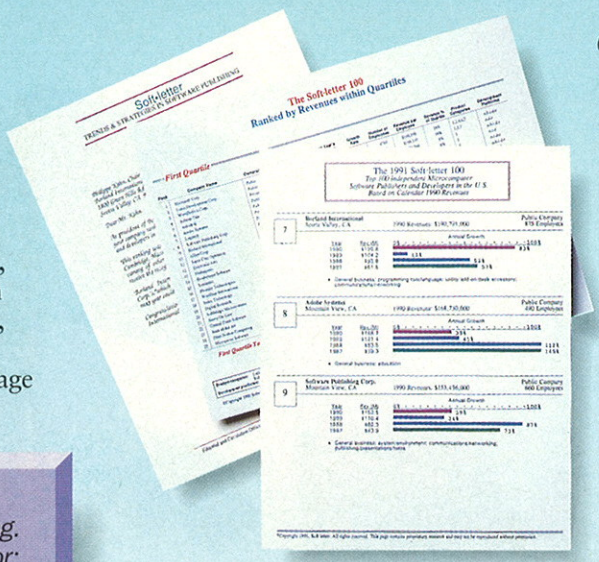
*National telephone: Egham (0784) 430060 fax: (0784) 430050 International telephone: +44 784 430060 fax: +44 784 430050*

R&R Report Writer
NEW - WINDOWS CLIENT/
SERVER VERSION
Call 0628 788181 for more information!

# Database reports fast and easy!

***"(We) created a report in just four clicks of the mouse."***
***PC Week, December, 1992***

R&R Report Writer, the ultimate data access and reporting tool, enables you to select, analyse, summarise, and present data in any form you need – columnar listings, form letters, mailing labels, business forms, even multi-page analytical reports.




*Nothing beats R&R for ad hoc database reporting. Versions now available for:*

| | |
|---|---|
| ***Xbase (.dbf)*** | ***Paradox*** |
| ***dBASE*** | ***Worksheets*** |
| ***Fox*** | ***1-2-3*** |
| ***Clipper*** | ***Symphony*** |
| ***Alpha Four*** | ***Quattro Pro*** |
| ***Arago*** | ***Oracle SQL*** |
| ***Access*** | ***NetWare SQL*** |
| ***Approach*** | ***XDB SQL*** |
| ***PerForm*** | ***Platinum*** |
| ***SBT*** | ***Telemagic*** |

### *Simply the best for all-around report writing!*

R&R has all the features to meet your every reporting need. Relate and query data, sort and group information, use spreadsheet calculations and totals. You can even format your layout with different fonts to get polished reports that will make you look great!

### *Turn your raw data into useful information.*

R&R's clean, uncluttered interface makes it the most user-friendly report writer available today. Create any report you need, exactly the way you need it – without having to be an expert in database programming.

### *The professional strength report writer.*

R&R is easy-to-use, yet it's packed with features to satisfy even the most demanding application developers. Features like an unlimited-use, **royalty-free runtime version,** allow you to distribute reports freely or weave them into your applications–a real time saver!

***FREE demo disk. Call today!***

### *Over 150,000 users have made R&R Report Writer the industry standard...*

R&R consistently wins praise from reviewers, developers, and end-users alike. And, great reports keep coming in:

*"R&R is the best report writer for Xbase and may very well be among the best ever offered for databases."*
***BYTE, July, 1992***

*"...the product does exactly what it advertises and does it well."* ***DBMS, January, 1992***

INFO WORLD TOP 100 1989 · PC WEEK LABS ANALYST'S CHOICE · 1992 DATA BASED ADVISOR Readers Choice Award 1989-1992 · DBMS READERS' CHOICE 1989-1992

When you need database reports, R&R gives you the productivity you need every time, on time. At just £199, R&R for Xbase is unbeatable value.

**Call 0628 788181 today for a free Demo Disk!**

**Available From:**
**QBS SOFTWARE LTD**
**10 Barley Mow Passage**
**Chiswick**
**London W4 4PH**

**Tel: 081 994 4842**
**Fax: 081 994 3441**

**R&R Report Writer®**

**Concentric Data Systems, Inc.**
110 Turnpike Road, Westborough, MA 01581 · Tel: 508-366-1122 · Fax: 508-366-2954
In the U.K.: R&R Software Limited · Viking House, Denmark Street, Maidenhead, Berkshire SL6 7XJ · Tel: 44 (0) 628 788181 · Fax: 44 (0) 628 788182

# Some Thoughts on Pointers & Arrays

*Part 1: A C Perspective*

Both C and C++ make extensive use of pointers, though the way experienced programmers use them in the two languages is often quite different. Very early in most courses on C++ the concept of passing a value by reference is introduced. Almost always this is followed by an explanation of why strings (C type) are not passed by reference. The upshot of this is that the C++ novice is faced with a new technique and then told not to use it. Confusing? Well I think so.

The perceived problem lies with the way C supports arrays and then proceeds to provide string facilities via arrays of `char`. In this article I want to set you thinking about the C implementation of arrays. I do not intend to present you with a detailed explanation - I think that would take rather more space than our esteemed Editor would want to provide. What I want to do is to suggest some directions you might explore in order to get a better understanding of what is happening when you start passing arrays around your program. To start consider:

```
char ar[20];
char *c;
```

No prizes for guessing the type of `c`; but what exactly is `ar`? (Remember it is the identifier that has a type.) Strictly speaking `ar` is of type `array of 20 char`. The trouble is that this does not have any practical meaning when applied to C code. In practice it is treated as a variation on `char *`. The problem lies in the whole superstructure that C builds on this foundation.

The index operator `[]` is defined such that it dereferences a pointer via C's pointer arithmetic. We can create dynamic arrays; with arrays of pointers we can create storage efficient multidimensional arrays. Now the problem: since arrays are treated as pointers, an array of pointers is indistinguishable from a pointer to a pointer. So how does the pointer arithmetic work? Consider the following:

```
char *suit[] =
{
  "Club", "Diamond",
   "Heart", "Spade"
};
char *fixed[6][3] =
{
  "AC", "BD","EF",
  "GH","IJ","YX"
};
```

This creates an array of `char`. The task before you is to complete this code so that 'Diamond' will be displayed on the screen:

```
void test (?????)
{
  printf("%s",?????);
}

main()
{
  /* pass suit array */
  test(????);
  /* pass fixed array */
  test(????);
  return 1;
}
```

There are many possibilities which can be substituted for these queries. But can you find one that produces the desired result for *both* arrays? I will be fascinated to have an answer to this problem because it is a *real* problem in a *real* program which I have failed to solve. I need the first type of array for string literals, the second type for modifiable static storage and a potential third type for dynamic storage. All I can come up with is to abandon the second form and replace it with a dynamically assigned array, with a secondary array of `char *` to handle the individual data.

One of the design flaws in C is that we cannot explicitly state the first dimension of an array parameter; any attempt to do so will simply be ignored (ie `void fn (char c[5])` is converted to `void fn(char * c)` at compile time). However, if you really need to pass around fixed size arrays and have the type information checked you can do so by using a `struct` as a wrapper. For example:

```
struct T
{
  char x[20];
};
void test (struct T t)
{
  printf("%s",t.x);
}
int main()
{
  struct T t;
  strcpy(t.x,"This works");
  test(t);
  return 1;
}
```

Those of you who are still following this discussion will have realised that the prohibition on passing arrays by value has just been side-stepped. In cases such as this it would usually be more appropriate to pass the structure address via a pointer, preferably type-qualified with `const` unless you intend to change an item in the array. The code now reads:

```
void test (const struct T * t)
{
  printf("%s",t->x);
}
int main()
{
  struct T t;
  strcpy(t.x,"This works");
  test(&t);
  return 1;
}
```

Painful though it may be, in C, fixed size arrays have to be handled by different mechanisms from those used to manage arrays of unknown size at code writing time (though they may be known at compile time). For one-dimensioned arrays we can manage if we are willing to take the risks implied by letting C treat arrays as pointers.

Note that, while a defensively written function can detect the passing of a null pointer, it cannot determine that the programmer has passed an address of a plain variable when an array was needed. I strongly advocate that the true size of the array is also passed as a parameter so that such accidents are not likely to happen without detection (ie the prototype reads `void fn (T t[], int n)` rather than `void fn(T t[])`). I must confess that I am disturbed by the amount of code that crosses my desk which contains prototypes such as:

```
void fn(T *t);
```

when the programmer knows that `t` should be an array. Just because the compiler treats arrays as pointers is no reason for the programmer to write them that way. `void fn ( T t[])` is a safer prototype. Next time I will try to take a C++ programmer's perspective on this topic.

EXE

---

# Sam revisited

*Peter Collinson continues his exposé of the SAM editor, with a voyage through its command language.*

Last month, I talked about the visual aspects of Rob Pike's `sam` editor. This month I am going to continue the story and describe the command language that you use to make more than simple edits with `sam`. Let's recap a little, so I can put the later stuff in context.

The `sam` editor was designed for use with Plan 9 and has been ported for use with X. The editor has two sections: the editor part and the screen handling part. You can run the editor part on its own, but I have never done that. In normal use on an X screen, the editor consists of two coöperating processes: one handling the screen, keyboard and mouse; and the other dealing with the editing of the file. Figure 1 shows an editing session, I have reduced the size of the window for printing purposes. I have started the editor by saying:

```
sam xcal_edit.c xcal_memo.c
```

This has thrown a window up on the screen within which you can then create smaller editing windows. The bar at the top of the screen is added by my window manager, `twm`. In this example the screen area is split into two: the topmost window is the *command* window used for typing commands and displaying status messages. The bottom window is the contents of the file `xcal_edit.c`. We'll see what happens to the other file later. You need to look at last month's article for a full description of how the visual interface works.

## The Command Window

Many people use `sam` without recourse to the command language. However, there are often repetitive or complex editing tasks that are best done by careful application of commands. You can crawl through the file making changes with the mouse and keyboard. You can do this in *any* editor, but it's prone to error. You will often miss something that you wanted to change. It's also time consuming, since you are wasting your eye/brain power in doing something that the computer can do better.

The commands typed into the command window refer to the current file, selected by moving the mouse cursor into the window and clicking with the left button. Input from the keyboard will now go into that window. All `sam` windows are 'click-to-type'. Simply moving into a window won't make it current: you must press the left mouse button.

To type commands, you need to move into the command window, again by moving the mouse and clicking the left button. The commands that you type here will apply to the last file that you were using. This can sometimes be counter-intuitive, as it isn't immediately obvious which file the commands will effect.

The command window contains a history of the commands that you have typed. Moving the cursor to the last line, typing a command and pressing return sends the line from the screen handler process to the editor process. The editor changes the file contents and updates the appropriate window. No big deal.

What's unusual is the ability to edit the text in the command window above the current input line using the normal cut-and-paste techniques that are used elsewhere in the editor. When you edit the previous history, the changes stay in the screen handling process until you explicitly send them to the editor process.

So the command window has an 'active' line where the commands that you type will make something happen in the editor. If you edit above the line to make a new command, then you invoke a command by *sending* it, using the menu called up by the middle mouse button. A demonstration is given in Figure 2.

I have selected the command window and replaced the / in the search command by a question mark, creating a reverse search command. The replacement is done by sweeping out the /

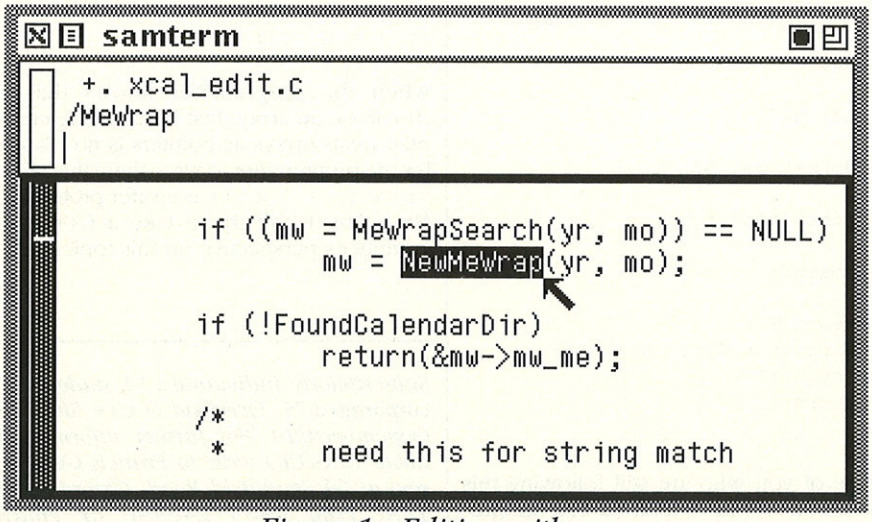
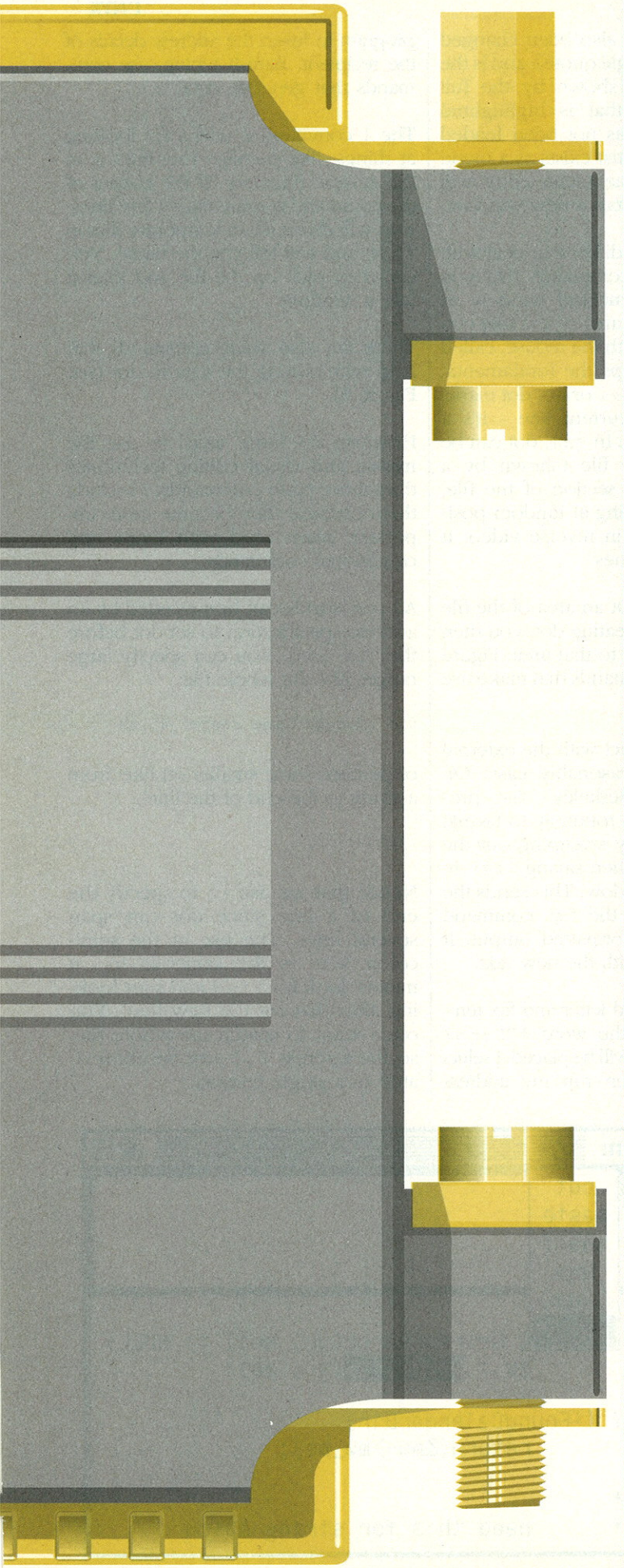



*Figure 1 - Editing with* `sam`



# Half the cost NOT half the dongle

Softlok II Keys, developed by Softlok International, have been designed to be an effective yet low cost solution to software piracy. Designed and manufactured in the UK for software developers worldwide, Softlok II contains all the flexibility you will need to protect your software.

Softlok II units are sophisticated keys containing 240 bytes of secure non volatile memory. An 8 byte password provides read/write protection. Both the memory area and password can be changed using the supplied routines from your application. As a programming adaptor is not required, they can also be altered in the field.

Our low cost starter pack contains everything you need to get you quickly up and running with routines provided on the disk for most DOS and Windows compilers. We can usually ship SoftlokII Keys for next day delivery so that you can get your software on the market just as quick.

**Low Cost** (less than half the cost of some dongles)

| | |
|---|---|
| Starter Kit | £20.00 |
| 1-19 Units | £16.90 |
| 20-49 Units | £15.40 |
| 50+ Units | £13.70 |

*All prices exclude VAT and delivery*


**Softlok International Ltd**
Imex House
40 Princess Street
Manchester
M1 6DE
England
Tel : 061 228 7379
Fax : 061 236 6890

**Softlok International Ltd**
Nieuwezijds
Voorburgwal 353
1012 RM Amsterdam
Netherlands
Tel : 020 638 3126
Fax : 020 638 3171


Member of FAST
Federation Against Software Theft

and typing ?. I now select the created command and click on the middle button to get the edit menu. I choose send to enter the new command and start the reverse search. This sounds complicated, but isn't once you get used to it.

The send operation always transmits a whole line from the terminal process to the editor process. You create a whole new command, sweep it out and send it. The new command is echoed at the bottom of the command window as if you had typed it in yourself. This differs from common usage with history editing in shells, where you tend to build up the new command piece by piece on the prompt line.

The command window behaves a little differently from the other windows in the editor, even though it looks substantially the same. At first, it feels a little odd to have a window on the screen that looks the same as the rest but behaves differently. The command window is 'more equal than the others'. It is perhaps a weakness in the user interface. The point is that you know how to edit text in the command window; nothing special is needed to create the new text. After a little time, you begin to wish that you had this ability in your terminal emulator.

## Basic commands

The commands for sam have their roots in the ed editor (and so will be familiar to ex or vi users). Some commands are very similar.

All these commands will print a result in the command window telling you that something has happened. Typical output from a normal start-up sequence is shown in line one in the command window in Figure 1. You see the name of the file that has been loaded, a full-stop to show that the file is current and a plus to show that the file has been read into the editor. This line is also added to the file menu that you can see by selecting the right mouse button.

Figure 3 shows the menu from the same editing sequence. Here we see the reference to xcal_memo.c that we entered on the command line. The bottom three lines refer to files: ~~sam~~ is the command window, xcal_edit.c has been loaded into the editor (shown by the plus sign in the menu); it has also been changed (shown by the single quote); and is the current window (shown by the full stop). The line that is highlighted xcal_memo.c has not been loaded into the editor, it just exists as a menu entry. Selecting the menu entry will load the file and start a new window.

Unlike the other editors that deal with line ranges, the commands that you type into the command window in sam deal with the area of text that you have swept out with the mouse, called *dot*. This is perhaps the fundamental change in sam. In vi or ed, dot marks the start of the current line - it's a position in the file. In sam, dot can be a position in the file (shown by a vertical bar) or a section of the file, starting and finishing at random positions and shown in reverse video. It can include newlines.

You can sweep out an area of the file with the mouse creating dot, you then apply a command to that area. Figure 4 gives some commands that make use of this idea.

Making sam interact with the external environment is reasonably easy. On my Sun, I have Berkeley's fmt program, which I use routinely to format text paragraphs by sweeping out the desired area and then saying |fmt in the command window. This sends the swept out area to the fmt command and captures its formatted output. It will replace dot with the new text.

I also have standard letter and fax templates containing the word ADDRESS where the address will be placed. I select this word and then run my address program to insert the address details of the recipient. Banish typing, use commands that generate data.

The ! command is useful for looking at things, like perhaps running ls in the current directory. If the output of the command is more than a few lines, sam will divert it to a temporary file on /tmp and will tell you the name. You can then pick up the file and load it into a window.

There are also some commands that help with editing the data in dot (see Figure 5).

However it's more usual to use the mouse and visual editing techniques than these plain commands. I include them because they become more important when used with loops and conditional commands.

All commands can be preceded by an address specification to set dot before they are used. You can specify large ranges like the whole file:

```
0,$ or some line range: 45,89
```

or perhaps some smaller set like 'from a string to the end of the line':

```
/fred/;/\n/
```

Notice that we use \n to specify the end of a line since dot can span several lines. The use of the semicolon here is the same as vi, it means 'look for fred and start looking after that for the new line'. You often want to search the whole file, so the address 0,$ can be abbreviated to a single comma.

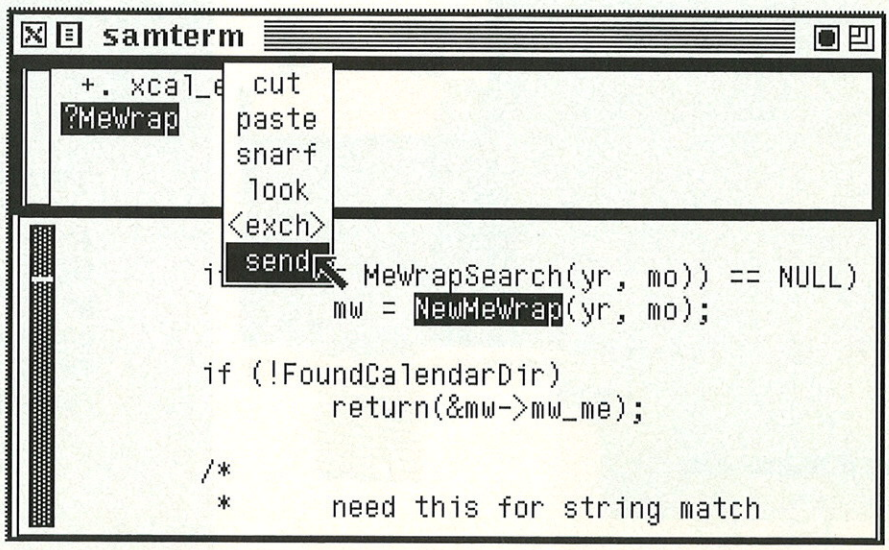



*Figure 2 - Editing the command window and the edit menu*

Typing a line address will make you go to that line, as you might expect. Hitting return or a p command displays the current line in the command window and sets dot to it.

## Loops and conditionals

A very common editing need is to make some repetitive change to the file. In sam, we set dot to some range and execute:

```
x/regexp/ command
```

This finds all the matches in dot for the regular expression and then applies the command to them. When the command is executed, dot is set to the matched text. To change all the occurrences of 'black' in a file to 'white' you would say:

```
,x/black/ c/white/
```

Remember that the comma is a shorthand for 0,$, so initially dot is set to the whole file. The file is searched for the string 'black'; when it is found dot is set to it and the text changed using the c command. I have separated the commands by a space, although this is not needed. The y command is the complement of the x command: it applies the command with dot set to the text between the matches.

The x and y generate loops. There are a pair of conditional commands that work with them. The g (for *guard* and not *global*) has a similar syntax to the x command:

```
g/regexp/ command
```

This runs the command exactly once if dot contains a match for the expression. It differs from x which runs the command for each match. The g command tests only for a match, and runs the command *without* changing the value of dot. The v command is the inverse of g, running the command if a match is not found.


*Figure 3 - The file menu*

| | | |
|---|---|---|
| < | *Unix command* | Replaces dot with the standard output of the command |
| > | *Unix command* | Sends dot to the standard input of the command |
| \| | *Unix command* | Replaces dot by the output of the command whose input is dot |
| ! | *Unix command* | Runs the command |

*Figure 4 - Examples of commands which can use dot*

The commands can be chained:

```
, x/.*\n/ g/Baggins/ p
```

will print all the lines in the file that contain the word Baggins. Reading from left to right: the comma sets dot to the whole file; the x command splits the file into lines, setting dot to each of them before invoking the next command; the regular expression here reads: 'any character except newline ('.'), repeated zero or more times ('*') followed by a newline (\n)'. So the g command is run with dot set to each line of the file. It will invoke its command if it finds the word Baggins in the line. So the chain of commands will print all the lines containing Baggins.

If we wanted to search for every reference to a Baggins except 'Frodo Baggins', then

```
, x/.*\n/ g/Baggins/
            v/Frodo/ p
```

would do the trick. Well, nearly, it would fail if 'Frodo' appeared on one line and 'Baggins' on the next.

The notion of commands chains is powerful and easy to use. It's simple to get hold of the idea that you extract data from the file by setting dot and then progressively reduce dot to the object you are looking for.

Of course, the key to success is a good understanding of the regular expression syntax for matching strings. The syntax of regular expressions in sam differs in some ways from that used in the older editors vi and ed. The expressions in sam are a rethink on the way that things work, largely because they deal with embedded newlines. Don't get me wrong, your knowledge of vi or ex regular expressions will port into sam. The differences are subtle but are easily learned. They are well documented in the various papers that come with the editor, so I'll leave you to seek that out if you are interested.

## Multiple file support

I have already mentioned that you can load file names into the right-button menu without necessarily loading the files into the editor. This can be done from the command line by supplying sam with a list of files to edit. It can also be done using the B command. You type:

```
B newfile
```

into the command window. This opens the file called newfile in the current directory and allows you to set up a window that will display the file. You can give the B command a number of files:

```
B a b c
```

You will be prompted to open a window for the file c, but a and b will be just loaded into the menu. You can also pipe the output from a command into the B command:

```
B <echo *.c
```

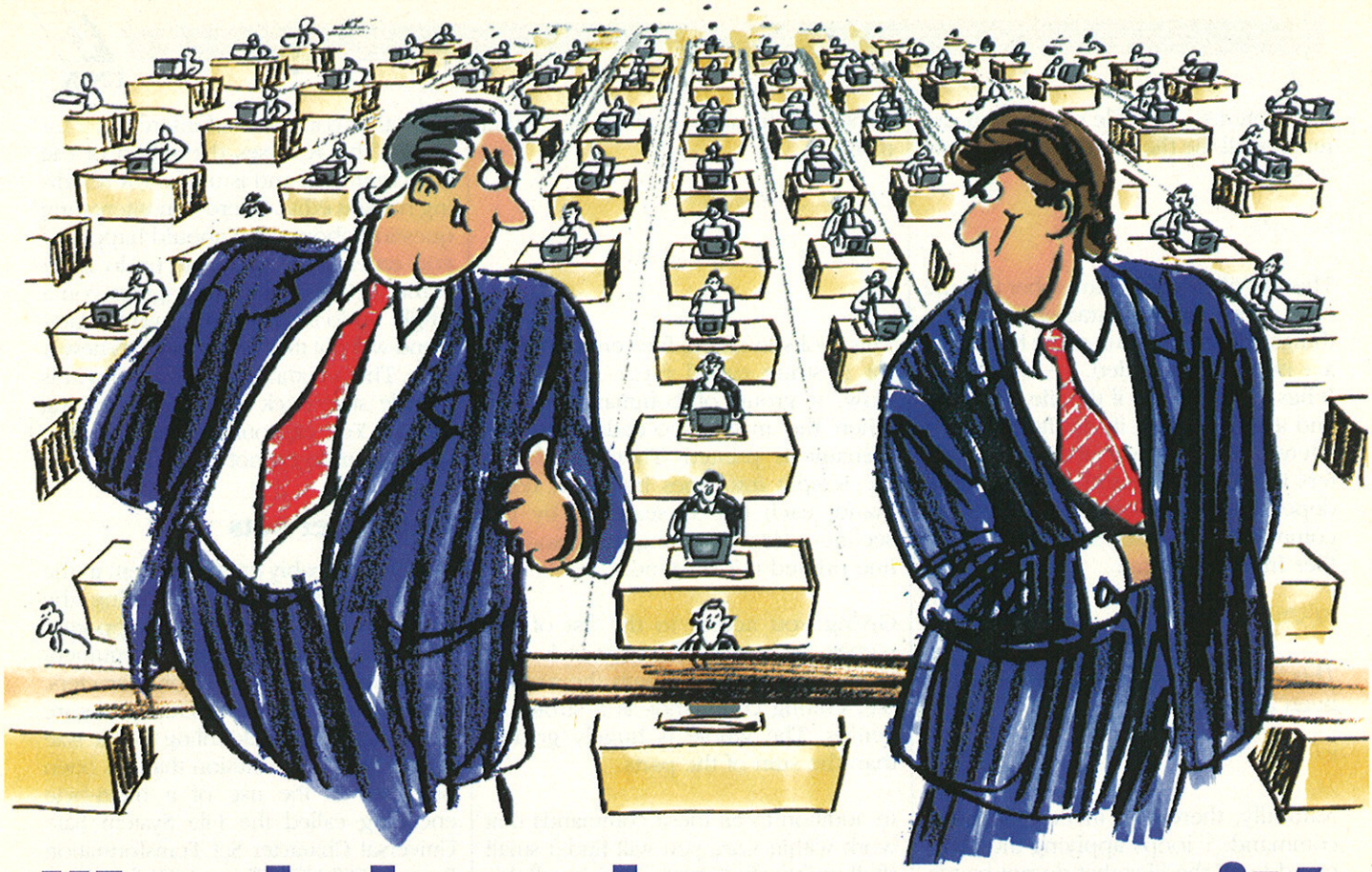
using the shell to create a list of files to be edited, or

```
B <grep -l look *.c
```

using grep with the -l option to generate a list of files that contain the string look.

The D command deletes files from the menu and from sam. Most of the time you do this from the close menu entry on the file menu, but a command is occasionally useful. Both operations don't touch the files on disk, just the references to the file held by sam.

The X command generates a loop like the x command, but the loop looks at



# "I only bought one copy – it's not protected with DESkey"


For years, the illicit use of software has been a major problem. It has been estimated that, in the professional environment, 55% of senior managers have at some time made an illegal copy of software. The solution to this problem is DESkey.

Designed and produced in the UK by our own engineers, with more than 14 years experience in the design of software protection modules (dongles), our current range are the most secure devices available.

- Customer unique seedable pseudo random number generation.
- High speed through encryption. Can also allow secure data storage.
- Up to 16,000,000 down counter for demonstration or 'pay as you use' purposes.
- Algorithmic functions prevent device driver emulation in multi process environments.
- Field programmability allows device redeployment.

**DESlock© for DOS**

A one line command encrypts any .EXE or COM file securely and quickly with no alteration to source code required, also supports user definable messages and regular DESkey checks. DESlock is also available for networks.

**DESlock© for Windows**

A one pass encryption system selected from an icon with no programming involvement required to secure the code. Very fast and very secure. A unique product only available from Data Encryption Systems.

- Available as transparent parallel, serial and bus versions for PC's and an intelligent RS232/423 device for any platform (UNIX, VMS etc).
- Drivers available in most major languages and environments including a DLL for Windows. ANSI C code is available for open systems.
- Full lifetime technical support.
- Free fully working evaluation kits.
- Designed and manufactured in the UK.
- Fast order turn around.

Member of FAST
Promoting the legal use of software

**Data Encryption Systems Limited**
*Silver Street House, Silver Street, Taunton, Somerset. TA1 3DL*

*Telephone 0823 352357*
*Fax 0823 352358*

the contents of the file menu. You can remove all the files that end in .c from sam with:

```
X/\.c/D
```

This has other uses. Remember that the menu has some characters that say things about the status of the file? It has a + if the file is loaded, a - otherwise. It has a . (full stop) if the file is current and a single quote if the file has been altered. We can look at these characters in the X command and do things depending on the state of the file. This command line deletes all the unread files from the menu:

```
X/-/D
```

This one writes all the files that have been altered:

```
X/'/w
```

Naturally, there's an inverse of the X command. Y loops applying the command to all the files that do not match the command. So:

```
Y/\.c/D
```

removes all the files that do *not* end in .c from sam. You can use these constructs in complex command sequences:

```
X/\.c/ , g/main/ f
```

Which says: for all the files ending in .c, look in them for a string called 'fred' and print the filenames that match (the f command). Notice how dot is set to the file contents and the g command is used to search the file. The g command searches dot, but doesn't change its contents.

You might also want to see where the matches occur. Here you can use the ability of sam to group commands to print the file name and the lines that match:

```
X/\.c/ , g/main/ {
    f
    , x/.*\n/ g/main/p
}
```

This looks in all the files ending in .c for a string called main as before. Now, a group of commands is run when the match succeeds. First the filename is printed. Then the whole file is split into lines by the x command, each line is searched by the second g for a matching string and the line printed if it is found.

Giving you access to the list of file names by a set of commands that mimic and interwork with the normal file, editing commands is a stroke of genius. The whole is hugely greater than the sum of the parts.

In addition to all these commands that work within sam, you will find a small shell script called B.sh. You install this somewhere in your search path as a command called B. You can now leave sam and wander about the file system, when you find a file or files that you are interested in editing, you say:

```
% B file1 file2
```

This sends a 'B' command to a named pipe on /tmp: sam is listening for commands to appear here and will execute the commands that you have sent.

## Miscellany

There are a couple more commands that prove to be useful. The first is k. This sets a mark at the current cursor position. You address the data at the mark by using the single quote character. Let's imagine that you want to write from 'here' to some other point in the file. I am assuming that the whole region that you want to write does not fit into a single screen.

First, you set the cursor to 'here'. You then say: k in the command window to set the mark to the current cursor position. You can now go off and find the end point of the region and set the cursor to that position. Now saying:

```
',.w file
```

will write the file region. Actually, this is not new. Setting and using a mark exists in vi and ed.

The other command of interest is u for *undo*. This reverses the action of the last command and is useful for repairing broken edits. There's an interesting question about what should happen if you enter more than one undo command. The vi editor only gives you a single undo step, a further undo command will put the file back into its newer state. The u command on sam unwinds the file state back through the editing history. You can only go back in time since an undo cannot be undone.

## Character sets

Plan 9 is probably the first system in the world to experiment with using the more complete character sets supported by Unicode. There is much argument about how to deal with 16-bit characters. Only the manufacturers of hard disks are really happy about doubling the file size of text files. One solution that has been proposed is the use of a multi-byte encoding called the File System Safe Universal Character Set Transformation Format, FSS-UCS-TF, or UTF for short. This is based on an 8-bit byte and uses a multi-byte encoding when needed. This is being pushed by X-Open. Plan 9 has generated a working version of this coding and sam has been programmed to use it.

Internally, sam uses a 16-bit character encoding called a *rune*. Externally, data is stored in sequences of 8-bit bytes. The good thing about the encoding is that it deals with ASCII transparently. If you only use 7-bit ASCII, the data remains compatible.

There is a slight problem with all this. If you type a control character like ^D into sam then it will be dealt with like any other character. However, your font must be able to show the control character as something visible, otherwise you will not see the character even if it is there. There are several X fonts that do this. I prefer to use the standard Sun screen-14 font, extending it to show control characters.

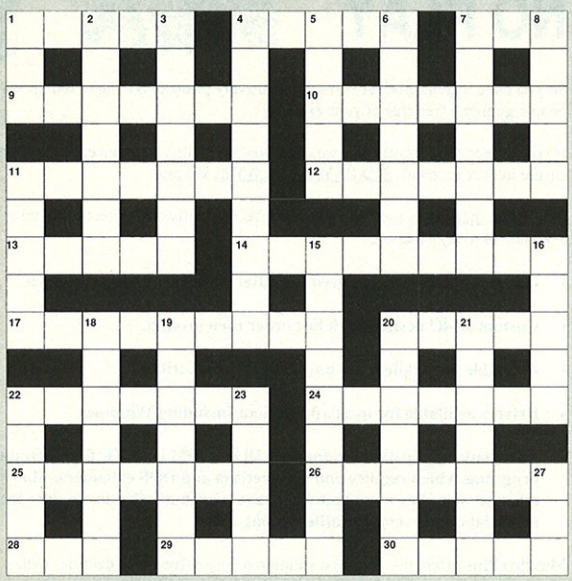
EXE

| | |
|---|---|
| a/*text*/ | Append text after dot |
| c/*text*/ | Change text in dot |
| i/*text*/ | Insert text before dot |
| d | Delete text in dot |
| s/*regexp*/*text* | Substitute text for match of regular expression in dot |
| m *address* | Move dot after address |
| t *address* | Copy dot after address |

*Figure 5 - Commands for editing dot*

*Peter Collinson is a freelance consultant specialising in UNIX. He can be reached electronically as* pc@hillside.co.uk *(although your mailer might be happier to put the address the other way round) or by phone on 0227 761824.*

# OCTOBER .EXEWORD



*.EXEword compiled by Eric Deeson*

***ACROSS***

**1** 1 dn or the result (5)
**4** The answer to discharge lies in the soil (5)
**7** Narrow bus that may get you to work? (3)
**9** Reflection concerning limit of 8 (7)
**10** George brings article to his 22 dn (7)
**11** Power of Victorian scientist and his time (7)
**12** Clever technique the French charge gently (7)
**13** About a hundred he puts in secret store (5)
**14** Quite amazed at the drunkard's walk (9)
**17** Freshwater alga beloved by school biologists (9)
**20** Trunk road returns to autumn month, so brown (5)
**22** Hard smart drink, maybe why he 14 (7)
**24** Carrier of signals between UK and the EC (7)
**25** Strange look from unsmiling star (7)
**26** Non-biological body part that may use IT (7)
**28** Prompt in the billiard hall (3)
**29** Sink of the Nile (5)
**30** Rubbery version of 27 (5)

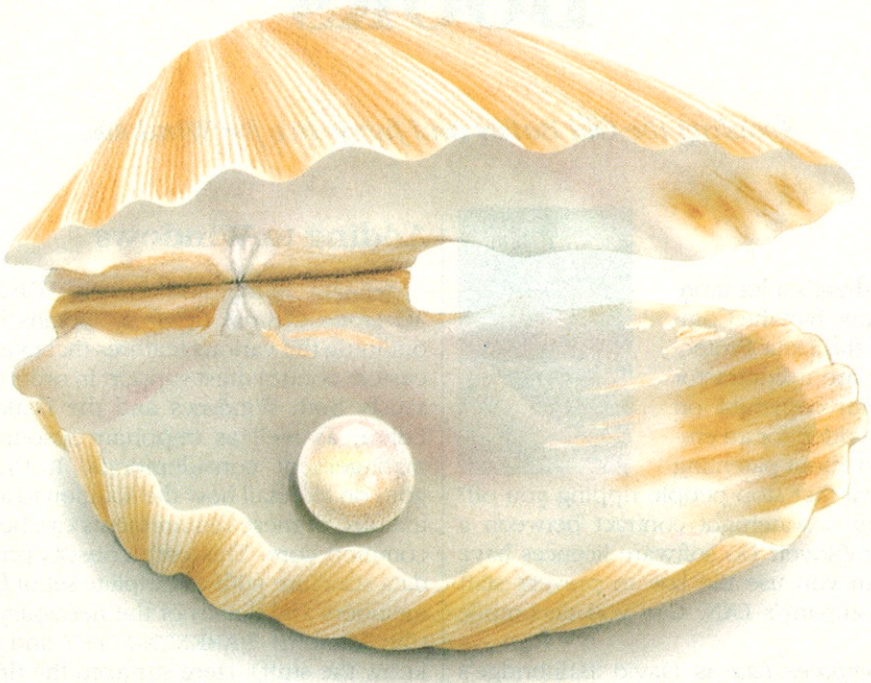
***DOWN***

**1** Go wrong with a message to say so (3)
**2** 5 takes chip to do with automation (7)
**3** Loud? Ear makes it rolled food (7)
**4** How a program passes through an eternal loop (9)
**5** Modern machine with brilliant start in main node (5)
**6** Sounding like factory time marker (7)
**7** American draught blows in spelling software (7)
**8** Set of peaks that defines the limits (5)
**11** Candle cores annoying people get on (5)
**15** Programmer (say) who follows no rules (9)
**16** Speak monotonously in hard raw language (5)
**18** Ask dbms in 25 sheets (7)
**19** Run the mainframe like a surgeon... (7)
**20** ... with his working tool (7)
**21** Hanging prettily perhaps (7)
**22** Logarithm chip is the basis of IT (5)
**23** Waiting prompt with 1 ac sounds shrill (5)
**27** Familiar American (from the South?) is for typesetting (3)

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R | E | M | A | R | K | ■ | D | E | C | L | A | R | E | D |
| E | ■ | N | ■ | O | ■ | ■ | F | ■ | O | ■ | O | ■ | O | |
| G | R | E | A | S | Y | ■ | O | F | F | S | H | O | O | T |
| I | ■ | M | ■ | T | ■ | ■ | I | ■ | I | ■ | K | ■ | A | |
| S | T | O | R | E | S | ■ | S | C | A | N | N | I | N | G |
| T | ■ | N | ■ | R | ■ | A | ■ | I | ■ | G | ■ | E | ■ | E |
| E | X | I | G | E | N | C | I | E | S | ■ | ■ | ■ | ■ | ■ |
| R | ■ | C | ■ | D | ■ | C | ■ | N | ■ | E | ■ | A | ■ | A |
| ■ | ■ | ■ | ■ | ■ | B | E | N | C | H | M | A | R | K | S |
| A | ■ | P | ■ | C | ■ | S | ■ | Y | ■ | U | ■ | G | ■ | S |
| P | R | O | G | R | E | S | S | ■ | C | L | A | U | S | E |
| A | ■ | L | ■ | E | ■ | T | ■ | ■ | A | ■ | M | ■ | S | |
| T | R | A | N | S | M | I | T | ■ | S | T | R | E | S | S |
| H | ■ | N | ■ | O | ■ | M | ■ | ■ | E | ■ | N | ■ | E | |
| Y | O | D | E | L | L | E | D | ■ | E | D | I | T | E | D |

***SEPTEMBER .EXEWORD***

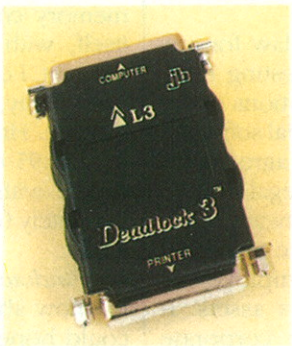
# When you look inside...



# You'll see that it's a real pearl of an idea.

We think you should know exactly what you're buying. That's why we've made sure you can look inside DeadLock 3.

DeadLock 3 gives you the best software protection available. Our unique combination of custom-designed ASIC and proven mathematical algorithm lets us safeguard your product without hiding anything. And that means you stay in complete control.DeadLock 3 protection is dead easy to implement too. Our user-friendly utilities make it a snap to develop the scheme that's just right for your product. And DeadLock 3 is so secure, we're even ready to give you the complete sources*.




Contact us today and find out about DeadLock 3. We've got a demo kit waiting for you - complete with utilities, samples and full documentation. There's absolutely no obligation. We just want you to look inside and see what a pearl of an idea it is.

**Deadlock 3™** *is real protection*

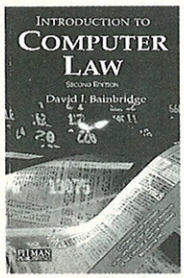
**BL Computer Security Limited**
**101 Hendon Lane, Finchley, London N3 3SH**
**PO Box 2543, London N3 3UA, United Kingdom**
**Fax: 081 346 2672 Tel: 081 343 0734**

*Sources available in compliance with terms specified by the manufacturer

# Books

*Avoiding the Old Bailey and furry dice for Windows.*

## Is IT Legal?




Behind the scenes, there's a lot more to computers and the law than the Computer Misuse Act and the Data Protection Act. What if you are a contractor and you want to claim copyright on programs that you've written for an employer? How do you go about patenting your programs, and will that stop people ripping you off? How do you draw up a watertight contract between a programmer and an employer? Do software licences have any legal strength? Can you use the law to prevent staff from misusing your company's LAN? Can you decompile commercial software?

*Introduction to Computer Law* is David Bainbridge's attempt to answer all these questions, and more, in a text that's free from legal jargon and readable by computer professionals. In this he succeeds admirably, producing a book that can be read all in one go, as a general backgrounder, or be kept as a reference work in case specific advice is needed.

As someone who writes a great deal about computer security, I've ploughed through a large number of legal texts and statutes in my time. It's refreshing to find something here that uses no legal jargon, yet still manages to explain the legal points extremely well.

Actually, I lie: there is a small degree of legal jargon, but it's all explained thoroughly.

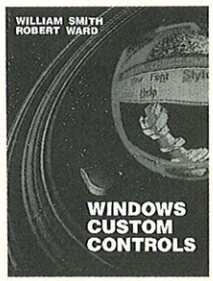
Even if you're not interested in computing law for any commercial reason, the book is still an interesting read. There are, you will discover, a number of situations under which you are legally entitled to copy commercial software. And despite what the small print on the licence agreements tell you, there are cases in which you have a legal right to disassemble and/or decompile commercial software.

*Introduction to Computer Law* should be essential reading for any programmer, contractor, employer or software company that wishes to protect its assets. It should also be studied carefully by those corporate lawyers who insist on creating software licence agreements which are so poorly worded that even Oliver Reed would have a better chance of standing up in court. Allegedly.

**Robert Schifreen**

*Title: Introduction to Computer Law* — *Price: £18.95*
*Author: David Bainbridge* — *Pages: 245*
*Publisher: Pitman Publishing* — *ISBN: 0-273-60106-7*

## Adding to Windows




William Smith and Robert Ward's *Windows Custom Controls* begins by outlining the various features that every custom control must support in order to satisfy both Windows and the Dialog Editor, as well as important resource management considerations. It then goes on to detail how the solution to all, or at least the most tedious, of these requirements can be encapsulated in one common core. This core consists primarily of a common data structure with a complete set of functions to access its members and details of the necessary messaging interface (the usual `WM_PARENTNOTIFY` and `WM_CTLCOLOR`, you know the stuff). Here surfaced the first of only two gripes I had with the book: the authors define the responses to `WM_CTLCOLOR` differently from the Windows norm. Any deviation from Windows' orthodoxy always worries me. It usually leads to all sorts of problems later as you inevitably have to fight to impose your heresy upon the orthodox might of Windows. I can't see that this would be the case here, but...

The bulk of the book builds on this foundation to develop a number of different types of custom control, starting with a versatile static (output only) control and moving on to dynamic (input/output) controls, various buttons, hybrid controls, sub-classed controls and virtual memory controls. Each of these groups has a chapter to itself, with the full code listings at the end. Brilliant, I thought, I can get straight into custom control development right away, simply by copy the source from the *supplied* diskette. Hmmm, no diskette: you would have thought that for £31.95 the publisher could have shipped a disk of the source with the book. Unfortunately you have to order it separately (or, like me, type in the code by hand). My second gripe.

*Windows Custom Controls* provides everything you need to know about custom controls and related topics that you could hope to find in a single book. I received the book on Saturday and by Monday I was using it, in anger, on a real, fee-paying, project. What better testimony could I give than that? Go and buy this book (and its companion disk) now.

**Edward Kenworthy**

*Title: Windows Custom Controls* — *Price: £31.95*
*Authors: William Smith and Robert Ward* — *Pages: 530*
*Publisher: Prentice-Hall* — *ISBN: 0-13-034497-4*

### Books Received This Month

| | | | | | |
|---|---|---|---|---|---|
| Discrete Event Simulation in C | Kevin Watkins | McGraw-Hill | £34.95 | 0-07-707733-4 | 384 |
| Constructing Logic Programs | Jean-Marie Jacquet | John Wiley | £19.95 | 0-471-93789-4 | 308 |
| Database Graphics Programming | Jason J Manager | Sigma | £19.95 | 1-85058-504-0 | 315 |
| Windows, Advanced Programming & Design | Peter J Morris | Butterworth Heinemann | £29.95 | 0-7506-0636-3 | 910 |

# ADVERTISERS INDEX

| ADVERTISER | PRODUCT/SERVICE | CIRCLE | PAGE |
|---|---|---|---|
| AP Software | Shareware | 106 | 78 |
| Bits per Second | Graphics Tools | 086 | 47 |
| BL Security | Software Security Dongle | 117 | 91 |
| Borland | Development Tools | 062 | 5 |
| Brent Communications | MAX Copy Protection | 077 | 33 |
| Butterworth Heinemann | Books Specialists | 110 | 85 |
| Clarion Software | Database | 070 | 21 |
| Contemporary I | BASIC Database | 076 | 31 |
| Contemporary II | Programming Tools | 082 | 39 |
| CTL | Copy Protection Hardware | 115 | 85 |
| DES | Software Protection | 111 | 87 |
| Evergreen | CASE Tools | 071 | 23 |
| Expotech I | Development Tools | 060 | IFC |
| Expotech II | Development Tools | 095 | 63 |
| Expotech III | Development Tools | 096 | 65 |
| Expotech IV | Development Tools | 097 | 67 |
| Expotech V | Development Tools | 099 | 69 |
| Forté Software Tools | Version Control | 093 | 24 |
| Great Western Instruments | Software Design Tools | 094 | 61 |
| Grey Matter | Programming Tools | 061 | 3 |
| Hoskyns | Development Tools | 085 | 45 |
| Hypersoft Europe | Programming Tools | 098 | 68 |
| Inmark | C++ Application Framework | 119 | OBC |
| Inst. Analysts/Progmrs | Institute | 114 | 85 |
| Intasoft | Version Control | 084 | 43 |
| IT Events | Windows EXPO | 090 | 53 |
| Kibworth | Training | 087 | 48 |
| LBMS | Seminars | 073 | 25 |
| Magnifeye | Software Protection Device | 088 | 49 |
| MAXPro | Security Products | 113 | 89 |
| Microcosm | Copy Control | 078 | 33 |
| Microsoft Press | Books Specialists | 079 | 33 |
| OS/2 Solutions Centre | OS/2 Software | 069 | 19 |
| PC Bookshop | Books Specialists | 105 | 78 |
| QA | Training | 118 | IBC |
| QBS I | Clipper Tools | 067 | 13 |
| QBS II | Clipper Tools | 083 | 41 |
| QBS III | Clipper Tools | 104 | 77 |
| R&R | Report Writer | 108 | 80 |
| Rainbow Technologies | Security Products | 089 | 51 |
| Readmar | Version Control | 074 | 27 |
| Real Techniques & Methods | CASE Tools | 075 | 29 |
| Richfords | Training | 064 | 9 |
| Ridgeway Press | Computer Manuals & Printers | 112 | 89 |
| Salford Software | Fortran for DOS & UNIX | 100 | 70 |
| Sequitter | Development Tools | 068 | 17 |
| Softlok | Piracy Protection | 109 | 83 |
| Software Security | Security Dongles | 107 | 79 |
| Staria | Telephony | 081 | 37 |
| System C | Application Generator | 091 | 55 |
| System Science I | Programming Tools | 080 | 35 |
| System Science II | Programming Tools | 103 | 75 |
| SystemStar SoftTools Ltd I | Programming Tools | 063 | 7 |
| SystemStar SoftTools Ltd II | Programming Tools | 065 | 11 |
| SystemStar SoftTools Ltd III | Programming Tools | 066 | 11 |
| SystemStar SoftTools Ltd IV | Programming Tools | 072 | 61 |
| Unicom | Seminar | 101 | 71 |
| User Friendly | Software Copy Control | 116 | 90 |
| Watcom | Development | 102 | 73 |
| Zinc | GUI Library | 092 | 57 |

# The code walkthru

*Sugar and fudge finding their way into program listings...*

'I want these pages clean, clean, clean. No rude four-letter words in any of *my* program listings. Lose the F-word, the B-word, the E-word. Wink Inc is a highly regarded software house. Our customers expect the best...'

'The E-word?' I quizzed, as I left her office. Phyllis Tyne, our new Technical Director. Joined us three months back, with 20 years 'cutting edge' development in PL/I under her spare tyre(s). Mizz Tyne, as she insisted on being called, was the chief developer of SOAP, a rather unique program from DIM Systems. (I believe the founders had a background in Basic. It's been DIM ever since.) SOAP was a tool for washing away rude words from code listings (as in 'wash your mouth with...'). Apparently, Mizz Tyne was responsible for filling the entries in the database of swear words. It's rumoured she went all the way up to Z, although noone has yet discovered her entry for Z.

Back at my desk, I fired-up WINCHAT, and dialled Dave. (It was Dave's code which had been the subject of Mizz Tyne's meeting.)

'Philistine has seen your code, Dave. And she doesn't like it. She doesn't like it at all.'

'It's not fair you know,' he grudged. 'She got me last time.

'Things aren't so bad, Dave,' I keyed tenderly. 'It's your colourful commenting. Let's have a look at what you've done.' What does that mean?'

'What does what mean?'

'Second line in `MyFunc69()`. The comment. Explain: *watch out for potential zif!*'

'Oh that,' he chuckled (implied in WINCHAT by the 'happy face' `:-)` symbol used on BBSs). 'That means a memory leak which grows and grows until it bursts, splatting your data segment with pus. I wanted to find the Z-word.'

'Yuck! She took particular dislike to the variable you declared in `WontWork()`. Why `WontWork()`? Come on Dave, what's the catch.'

With a beaming smile (several happy faces this time) he revealed his genius. 'That function will never compile. You see, the variable which Mizz Tyne so detests, will look like this - `ACompleteF***Up` - after SOAP has been at it.'

Two hours, three cups of coffee and 30 pages of listing paper later I made my way back to Mizz Tyne's office. 'Phyllis. Are you in? Here's the latest version of Dave's code listing. I hope it meets your approval.'

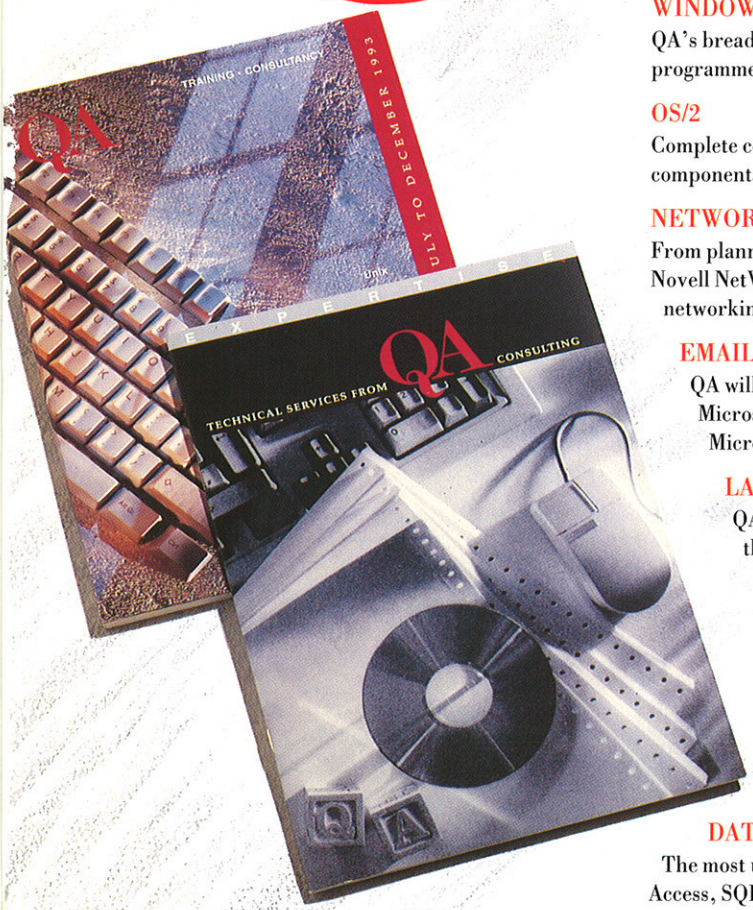
EXE

*In the course of writing this article .EXE acquired a new high tech tool, SOAP NT from DIM Systems. We are sorry for any misspellings which may have arisen as a direct result of its use. In particular, we would like to apologise in advance to Wink Inc (sorry, there I go again).*

TRAINING AND CONSULTING

# Skill up with the leaders

Computer professionals know that QA provides the best technical training and consulting, giving them skills in new technologies and keeping them up-to-date with the latest developments.




## UNIX

QA can help you make best use of Unix workstations or servers with Fundamentals, Administration, Networking and Programming courses.

## WINDOWS AND WINDOWS NT

QA's breadth and experience in Windows and Windows NT is unrivalled, with courses for programmers, support staff and users. QA is a Microsoft Systems ATC.

## OS/2

Complete coverage of OS/2 including programming and support, the extended services components, OS/2 LANS and even courses for first time users of the Workplace Shell.

## NETWORKS

From planning to implementation, support to programming, QA has extensive training in Novell NetWare (NAEC), Microsoft LAN Manager, IBM OS/2 LAN Server and Unix networking. Datacommunications training too.

## EMAIL AND GROUPWARE

QA will teach you how to provide an efficient service for users of Lotus cc:Mail, Microsoft Mail, and how to use the powerful groupware products; Lotus Notes and Microsoft Windows for Workgroups. QA is a training centre for Lotus CLE.

## LANGUAGES

QA is the UK's premier C and C++ training organisation. We have taught thousands of delegates good, structured, efficient C and C++ programming style.

## OBJECT-ORIENTED TECHNOLOGY

From introductory courses and management overviews to sophisticated analysis and design, including Rumbaugh OMT, QA provides a complete education in the principles and practice of Object Technology.

## SUPPORT

Manage PCs and workstations better with hardware and software support training for DOS, Windows, OS/2 and Unix.

## DATABASES and CLIENT-SERVER

The most up-to-date training in database, SQL and client-server technology. Microsoft Access, SQL Server and IBM DB2/2 support and programming are included.

We are dedicated to helping our clients get the most from their computer investment through quality training and consulting.

***For more information, call Gordon Ritchie on 0285 655888, or complete and return the coupon today.***

QA

Mail to QA Training, Cecily Hill Castle, Cirencester, Gloucestershire, GL7 2EF. UK.

Tick here for full details.

- ☐ TRAINING
- ☐ CONSULTING
- ☐ Unix
- ☐ Windows
- ☐ OS/2
- ☐ Networks
- ☐ Languages
- ☐ Object-Oriented
- ☐ Email/Groupware
- ☐ Support
- ☐ Client-Server/Databases

➤ CIRCLE NO. 118

Name _____

Job Title _____

Company _____

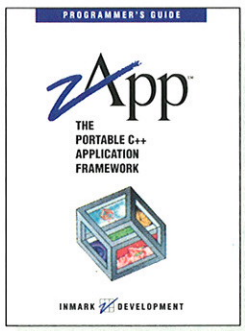
Address _____

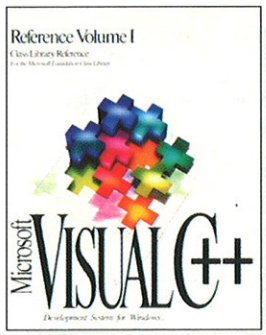
_____

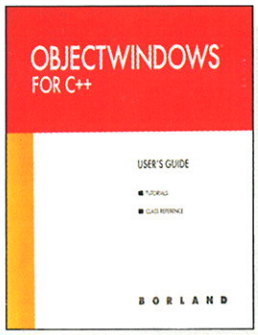
Postcode _____

Telephone _____

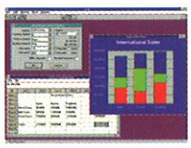
EXE 10 93



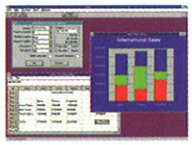




# VERY PORTABLE BARELY PALATABLE

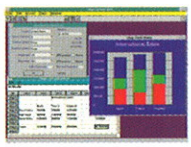


WINDOWS

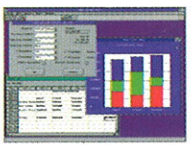


WINDOWS NT

OS 2.x



UNIX X/MOTIF



DOS GRAPHICS

DOS TEXT

Ever get the feeling that OWL stands for "Outdated Windows Library"? That MFC really means "Microsoft Frustration Classes"? If so, toss them out and join the thousands of programmers who have discovered a better application framework: **zApp**.

In an industry that's always changing, the future belongs to programs that can change with it. That's why we've developed zApp, the industry leading C++ GUI class library. zApp allows you to support all the platforms you need with just one set of source code. You simply make calls to zApp classes, without worrying about the platform's specific API, and zApp does the rest.

zApp is also the most full featured application framework available. Giving you more classes than OWL and MFC combined, zApp sports an unmatched list of capabilities for creating first-class applications quickly and easily. That's one of the reasons why Windows Tech Journal wrote "zApp is simply the best designed application framework on the market. If C++ is your language of choice, then you should be using zApp. Period".

Finally, zApp is easy to use. By providing an intuitive class structure and plenty of example programs, zApp allows you to produce more code faster, making it ideal for C++ programmers of all levels. With over 1200 pages of documentation, including extensive tutorials and demos, as well as free technical support, you know you'll never be left hanging.

So pick up the phone now and call **(0992) 500919** and ask for our free demo disk, and get a glimpse of what the future has to offer.

zApp

- **Source Code Included**
- **No Runtime Royalties**
- **Compatibility with Most Major Compiler**
- **Documentation** - including over 65 sample programs, tutorials and 880 page Programmers Reference
- **Over 35 Window Classes** – including application, Dialog, MDI, all Standard Controls, Common Dialogs, and access to custom controls
- **Data Entry Classes** - includes Text, Picture String, Numeric, Date, Time, Radio Group, Checkbox and List Fields
- **Graphics classes** – including Windows, Bitmap, Printer, Fonts, Pens, Brushes, Bitmaps, Colors, and Regions, Printer and Font enumeration
- **Object Persistence** – Using a streams interface, easily store objects and other data to disk and retrieve them
- **Advanced MDI Support**
- **Robust and Efficient Memory Management**
- **Advanced Message Handling**
- **Advanced Debugging Support** – traps many common programming errors - includes a powerful assert system
- **Dynamic Sizers** – Sizers provide automated window and graphical object layout management
- **Advanced Scrolling Classes**
- **Menus** - including standard, system, and popup
- **Full DDE Support**
- **Much, much more**

**INMARK DEVELOPMENT**

2065 Landings Drive, Mountain View, CA 94043



**SYSTEMSTAR**
**SoftTools Limited**

*1-3 Parliament Square Hertford SG14 1EX*
*Telephone: (0992) 500919 Facsimile: (0992) 554261*